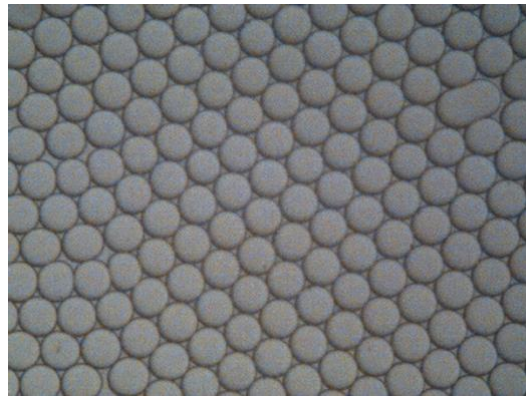
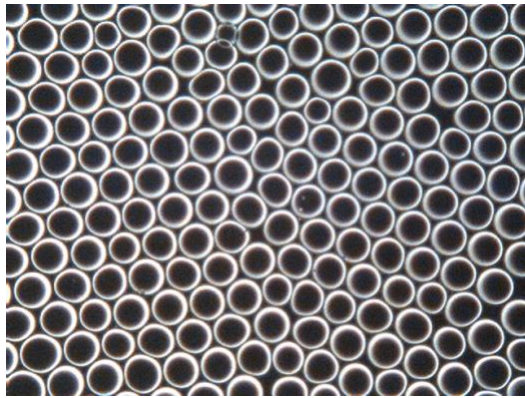


Home-made emulsions using MANATEE (MANifold Adapter and Tray for Easy Emulsions):

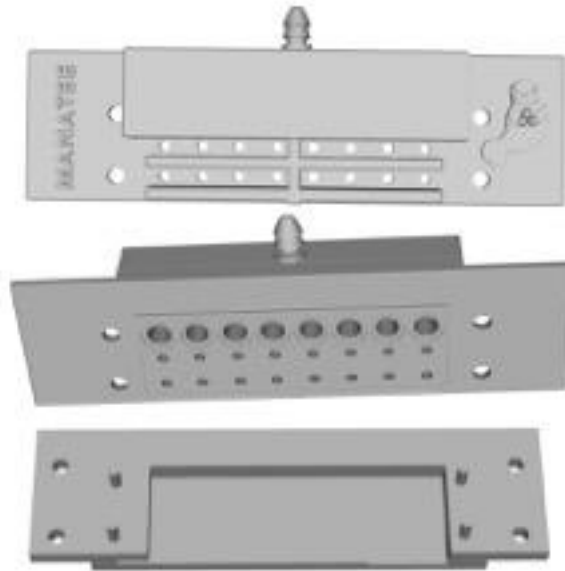


Phase-contrast and normal image of emulsions generated with MANATEE – these were generated using PBS with 1% BSA and BioRad EvaGreen oil.

Here are instructions for making uniform emulsions using a 3D printed manifold attached to a generic vacuum source and a simple Arduino-controlled vacuum release system. The system uses commercially available QX200 microfluidics chips from BioRad QX200™ Droplet Digital™ PCR System and rubber gaskets, and the EvaGreen Droplet Generation Oil. It can save you buying the droplet generator if you want to just make emulsions - we weren't interested in using it for digital droplet PCR so did not need the droplet reader either. This design could definitely be refined and improved, but as it stands it works just fine to make stable emulsions! If you have a very sensitive vacuum pump that can operate in the 5-15kPa range, then you can just attach that straight to the MANATEE and make your emulsions. We didn't have that available, so instead made a 'vacuum release' circuit which can be attached to a generic vacuum source and then just releases the low level of vacuum needed to generate the emulsions.

The MANATEE

The MANATEE manifold and tray were 3D printed using SLS with plastic polyamide 11: PA 11 plastic. 3D objects were uploaded to Shapeways (<https://www.shapeways.com/>) and ordered for 3D printing. See "3D_objects" folder for the 3D printing files. SLS was chosen since extrusion-based 3D printing tends to have a slightly ribbed/uneven surface and to allow the wells on the microfluidics chip to be completely sealed off I wanted a super smooth finish. The tray could be printed using standard extrusion-based 3D printing though since its function is mainly to keep the rubber gasket and the microfluidics chip secured. The MANATEE was designed so that the Manifold can be clamped onto the tray with a vice (or similar) and even pressure applied.



3D model of MANATEE – manifold shown from above and below, and tray with latches to attach rubber gasket.

The microfluidics chip only slots into the tray in one orientation. The oil is added to the bottom well, the mastermix to the middle well and the emulsion is collected from the top well.



QX200 microfluidics chip loading order

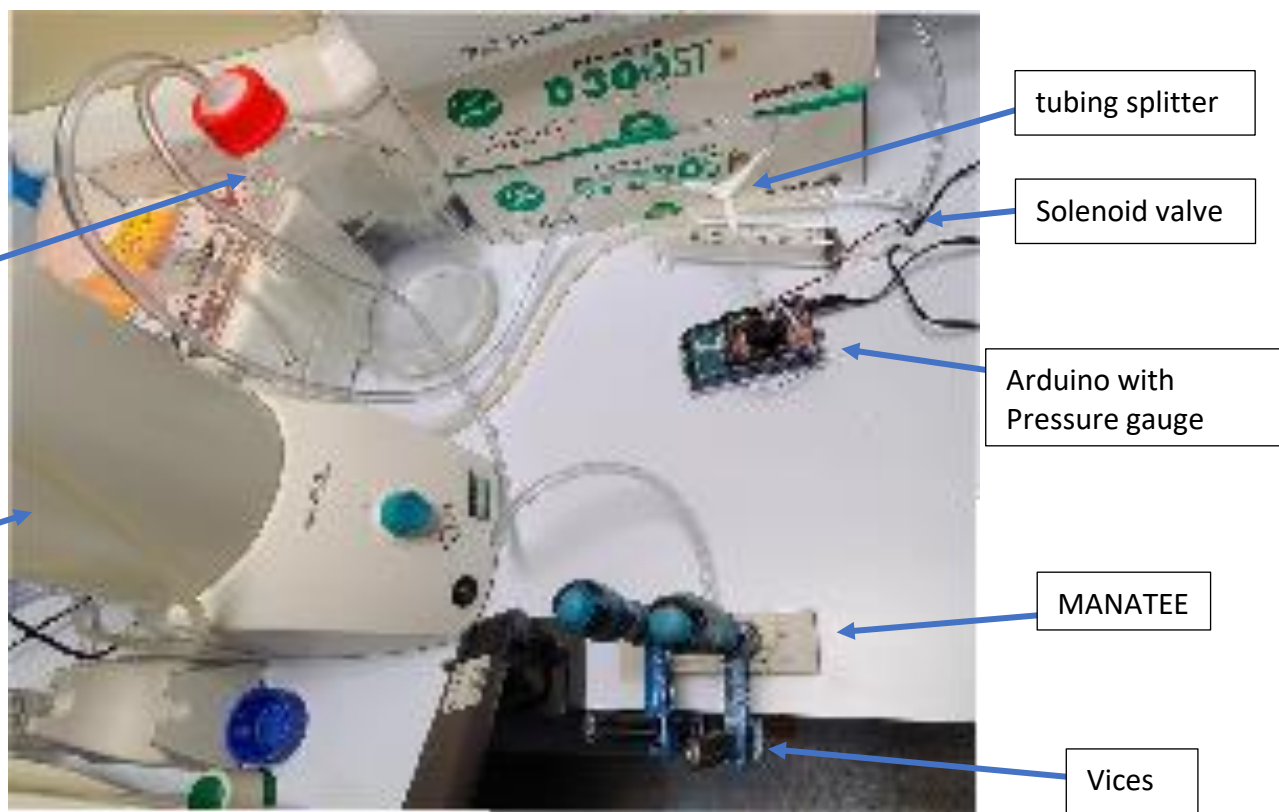
The gasket is the strapped over the chip on the tray, fitting the holes on either end of the gasket over the latches.

Then use a vice or similar to secure the manifold to the tray. I clamp both onto the workbench to ensure it is tightly sealed and no air enters the manifold while the droplets are being generated. Apply a vacuum (I've found 6-10kPa is generally the range needed for most PCR

reagents, but it will depend on the viscosity of the aqueous solution) until all the oil is drawn through the chip. You can see the level of oil left within the wells, looking at the manifold side-on.

Vacuum release system

If, like me, you don't have access to a vacuum pump that can apply very low levels of vacuum required for microfluidics, you can also build this vacuum release system. When looking for precision vacuum pumps (that we were allowed to buy via the university) I was looking at having to spend several thousand pounds. So instead I just thought I'd use our existing crude vacuum pumps (e.g. ELISA plate washer) and release vacuum from there in a controlled manner. The whole thing probably cost around £50 to make (including the Arduino and tubing etc.), although I did already have some of the components (like the Duran bottle and ports for the Duran Bottle lid and some of the tubing). The expensive components are the Arduino and the pressure gauge. This circuit could be moved onto one of the really small Arduino circuit boards since you only need one digital and one analogue pin. The image below displays the overview of the main components in the setup:

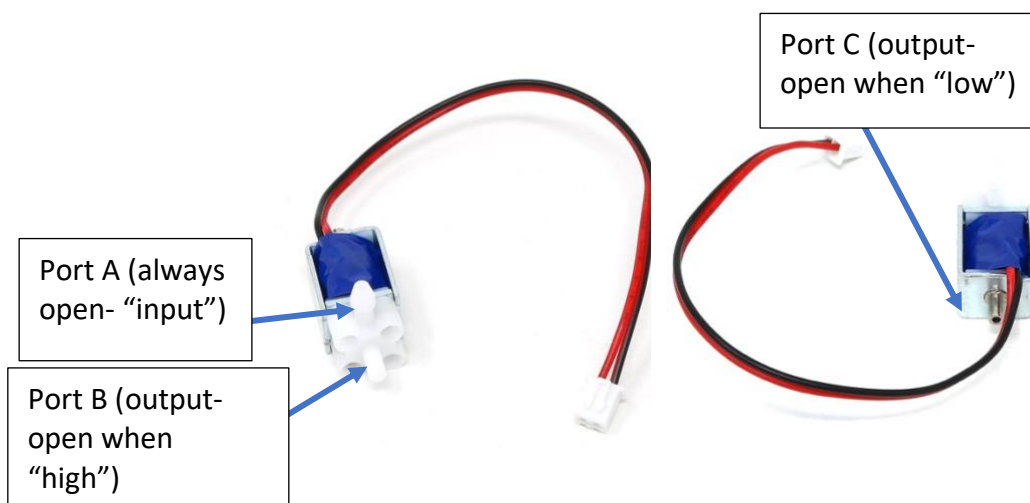


MANATEE and vacuum control system – final setup

Basically, the pressure gauge measures the vacuum in the system and if it is below the desired threshold, it briefly opens the solenoid valve to let more vacuum into the manifold, before closing off again. The circuit keeps opening and closing the solenoid valve to keep it roughly around the desired vacuum threshold. An LED light blinks each time more vacuum is released into the manifold, and as a way of making sure the circuit is working.

A brief note on the solenoid valve:

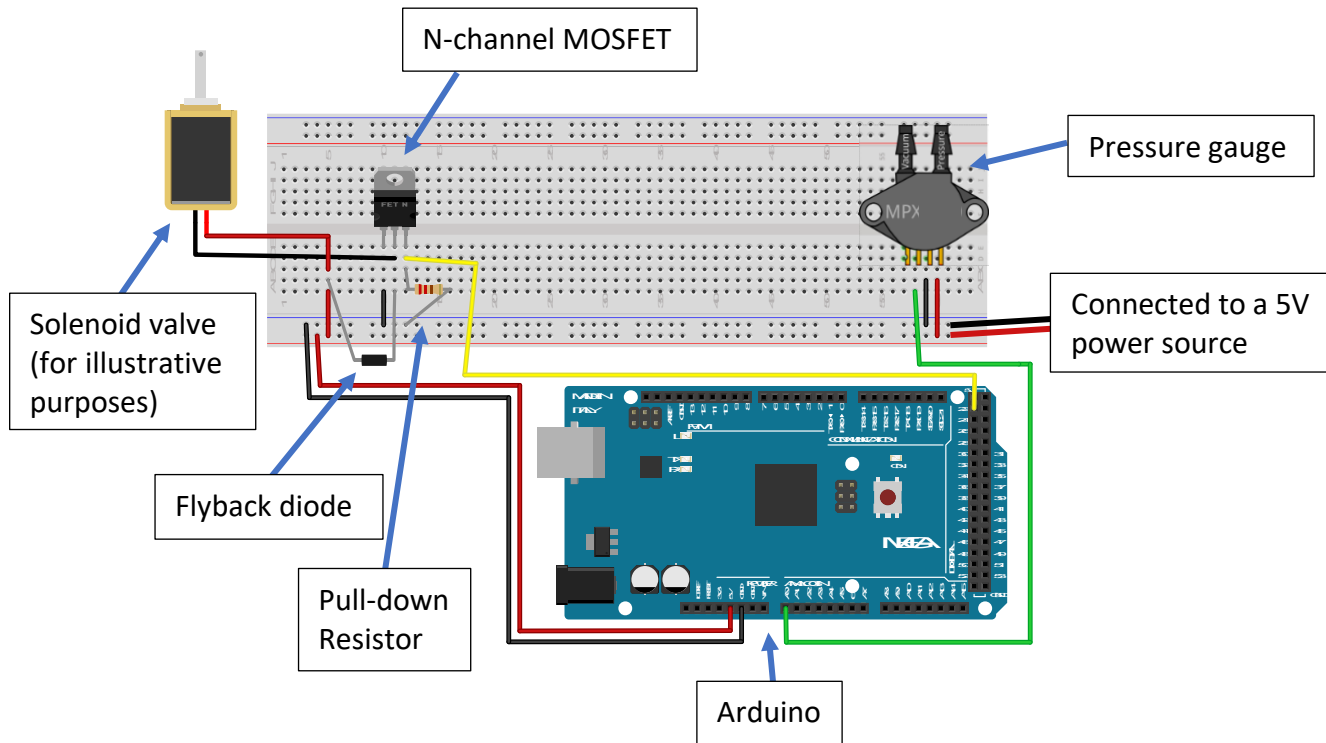
I used a solenoid air valve, after considering quite a few different options this was the cheapest and simplest I could find (<£3). It can also easily be replaced since it plugs into the circuit rather than being soldered in. However, it does have three ports and switches between the central port (Port A) opening towards the lower plastic port (Port B), and the upper metal port (Port C). Because of this, I sealed Port B off so that vacuum accumulates in the plate-washer pump / tubing which is then released into the manifold. In the images, I just connected port B to a syringe with some tubing because this was the easiest “stopper” available to me but anything would work.



(image from: <https://thepihut.com/products/6v-air-valve-with-2-pin-jst-ph-connector>)

Wiring up the circuit

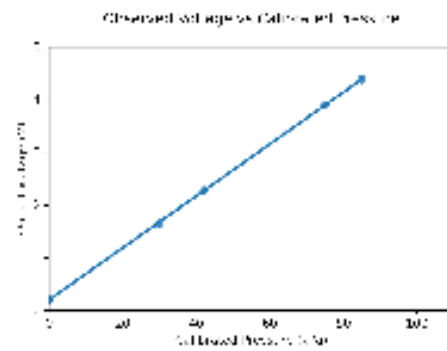
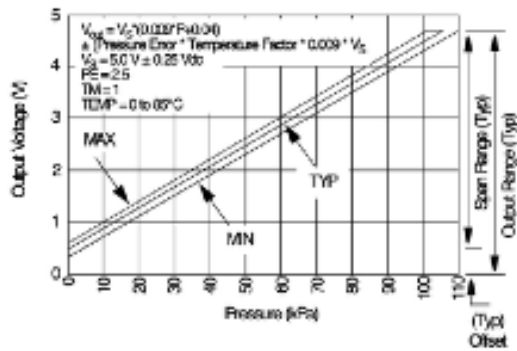
Here are the details for the circuit (shown on a breadboard – in the final version this was soldered to an Arduino prototyping board):



There are two main parts of the circuit: the pressure gauge and the solenoid. The pressure gauge is very simple:

I used an MPX5100 DP – a dual port differential pressure sensor (unlike the diagram it actually has 6 pins, but as in the diagram above, only the first three pins are used) which measures the difference in pressure between the two ports- one exposed to atmospheric pressure and the other one connected to the MANATEE.

In brief the MPX5100 DP pressure gauge is connected to an Analog input pin on the Arduino (A0 here) and it is connected to power and ground. Code for pressure gauge was adapted from here: <https://circuits4you.com/2016/05/13/arduino-pressure-measurement/> . As a quick sanity check that the calibration was correct, I checked the calibrated pressure against the observed output voltage and compared that to the manufacturer's datasheet.



Powering of the solenoid valve is controlled by a MOSFET, which in turn is controlled by the Arduino. The solenoid valve is connected to power directly but ground is connected to the middle pin of the N-Channel MOSFET. Pin 1 of the MOSFET connects to ground, pin 2 is the “output” connected to the solenoid and pin 3 is connected to the Arduino and to ground via a 220 ohm pull-down resistor. When the MOSFET receives a “HIGH” signal from the Arduino it opens the channel and powers the solenoid on, which blocks the airflow between the vacuum source and the MANATEE. When the MOSFET receives a “LOW” signal from the Arduino it powers the solenoid off and opens the airflow between the vacuum source and the MANATEE. A Flyback diode is included between the power and ground of the solenoid. If you want to include an LED to visually show whether the solenoid is powered on or off, you can include that between pin 3 of the MOSFET and the Digital pin 22 on the Arduino.

The final Arduino code can be found in the folder “Arduino_code”.

Connecting the tubes:

I connected the tubing as follows: The vacuum source was connected to Port A of the solenoid valve. Port B of the solenoid is sealed off and not used. Port C of the solenoid valve feeds into a three way tube splitter with one port leading to a large Duran bottle to buffer the vacuum which also leads to the MANATEE, and the other port leading to the vacuum sensing port of the dual-port pressure gauge. This allows the vacuum in the manifold to be measured.

To operate it, first turn on the vacuum release circuit, then the vacuum source.

Finally, it is worth mentioning that this system is pretty flexible: I’ve tried BioRad’s RT-PCR reagents and Roche Titan’s RT-PCR mastermix (with BSA added!) and gotten thermostable emulsions. Other PCR reagents work too, just titrate BSA and surfactant (I used a non-ionic plurionic surfactant) and try a range of vacuum levels. Some commercial PCR reagents produce emulsions that look OK initially but go “sticky” once thermocycled. I don’t know what additives in the mastermix make this happen, but I’d just recommend trying them out empirically and seeing what works.

Materials:

Part	Catalogue Numbers / specifications
Arduino	Arduino Mega 2560
Pressure Gage	MPX5100DP
Solenoid Valve	DFRobot 6V 2-Position 3-Way Air Valve for Arduino (DFR0866)
N channel power MOSFET (30V 60A)	IRLB8721PbF
Flyback diode	Vishay (1N4007E354)
Green LED	Kingbright 2 V Green LED 5mm (L-53GD, RS components)
Resistors	220 Ω Resistor (RS PRO
Prototyping PCB board	Arduino UNO R3 ProtoShield (483150)
Plastic PVC tubing	<ul style="list-style-type: none">- RS PRO Hose Pipe, PVC, 3mm ID, 6mm OD- 2.5mm inner diameter stretchable tubing (use tubing 2.5-3mm wide, depending on how you connect your vacuum source)
Y-tubing splitter	I used one which is compatible with tubes 2-6mm in diameter. This was quite helpful as I could stretch different sized tubes onto the same splitter.
Sealing grease	
Cable ties	