



The Quadrant Actuator Machine (QAM)

Pablo Vidal Franco

Updated January 29, 2026

Contents

1	Introduction	2
2	Electronics	4
3	Construction	4
4	Programming	5
5	Example dataset obtained using the QAM	6

The creation of the QAM was supported by the strategic programme Excellence Initiative at Jagiellonian University, Poland (RSM/122/FP).



1 Introduction

The Quadrant Actuator Machine (QAM) is an arena for experimental tests that consists of a removable Petri dish, four pull-type electromagnets and four quadrant-shaped hammers, separated by a margin and two walls to minimize vibration cross-talk. A PETG 3D-printed base holds the four electromagnets fitted with hammers. Another 3D-printed structure, (the crown holding the Petri dish), sits on top of it, held in place by a pair of 3D-printed grips (Figure 1). A prototype of the QAM was also built but it was discarded as its weight would not make it practical for transport (Figure 2). Table 1 shows the materials used during the construction of the prototype and the final version. I designed and assembled both the final QAM and the prototype. I also tested the machine on terrestrial isopods.

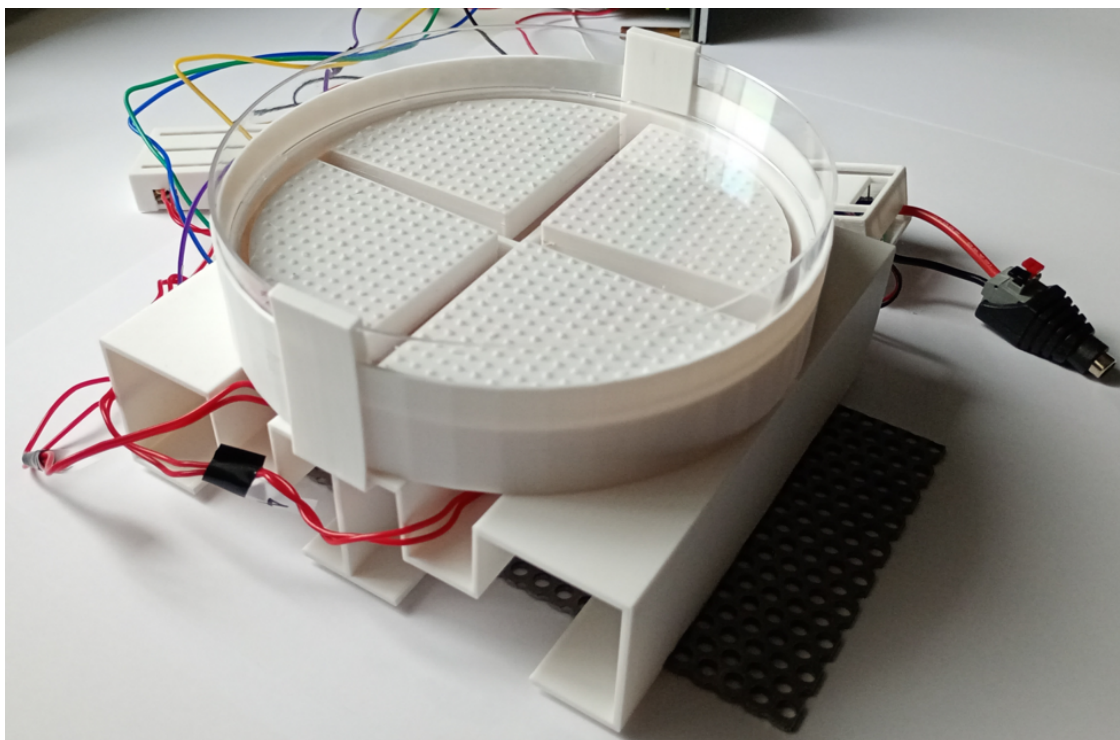


Figure 1: Picture of the final version of the Slim QAM.

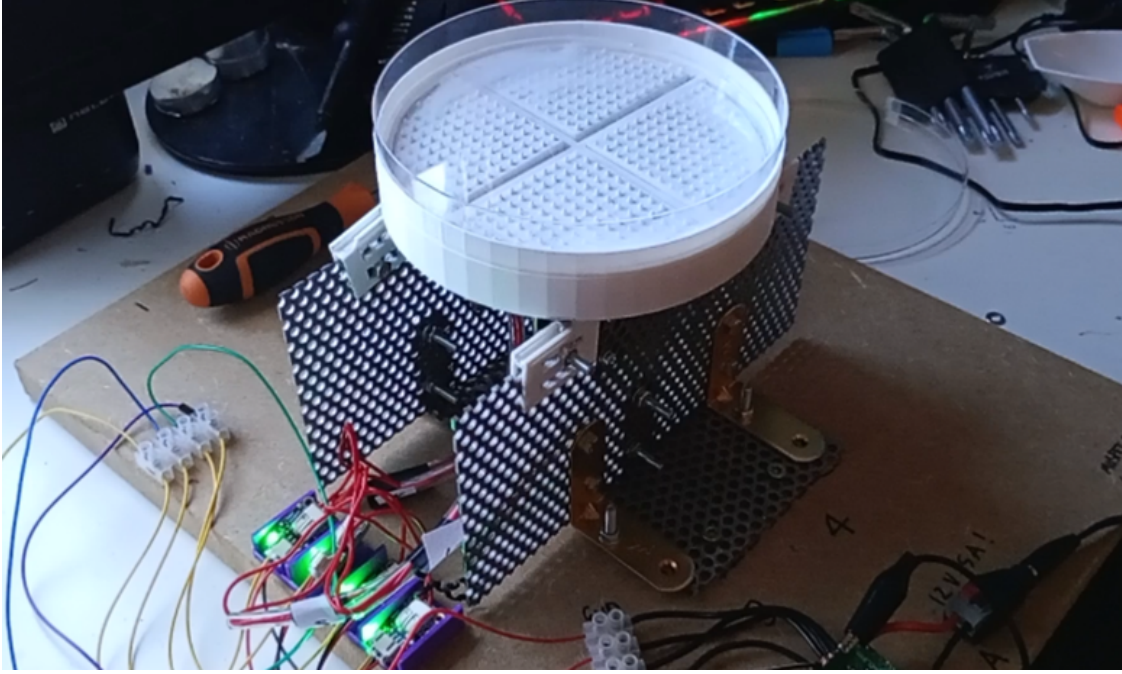


Figure 2: Picture of the QAM prototype.

Table 1: Materials used for the construction of the QAM prototype and the Slim QAM.

Item	Purpose	Number
140 mm diameter PET Petri dish	Removable part of machine forming the arena.	1
40 x 30 x 2 cm wooden sheet	Main support for the steel sheets and machine.	1
Perforated 20 x 10 x 0.2 cm steel sheet	Support for the electromagnets.	3
M3 screw	Attach the steel sheet to the wooden sheet.	4
L-brackets	Attach the steel sheets other steel sheets.	6
M3 bolt, nut and washer	Attach L-brackets to the steel sheets.	32
M1 bolt, nut and washer	Attach the 3D-printed pieces to the steel sheets.	4
Super glue	Bolt fixing in the prototype.	1
8 mm screw terminals 10A/250V	Wiring prototypes and final connections.	12
PVC wire set / wire		1
Power supply 12V/5A	Power supply for the electromagnets.	1
DC 5.5x2.1 mm power jack	Split the power into a PCB adapter.	1
Adapter PCB	Distribute (+) and (-) to electromagnets.	1
Raspberry Pi 5	Control the electromagnets and record data.	1
I ² C PWM driver	Interface to control the electromagnets.	1
N-channel MOSFET driver		4
12V 12W pull-type electromagnet	Produce vibrations on-demand (prototype).	4
12V 8W pull-type electromagnet	Produce vibrations on-demand (final version).	4
Solder roll	Solder connections.	1
PETG filament roll	3D-print prototypes and final versions of the hammers sitting on the electromagnets, and the Petri dish supports.	2

2 Electronics

Four pull-type electromagnet solenoids were wired through N-channel MOSFET drivers and an I²C PWM driver to be controlled by a microcomputer (a Raspberry Pi). Figure 3 shows the wiring schematic and Figure 4 shows the custom PCB designed to handle the four MOSFET units.

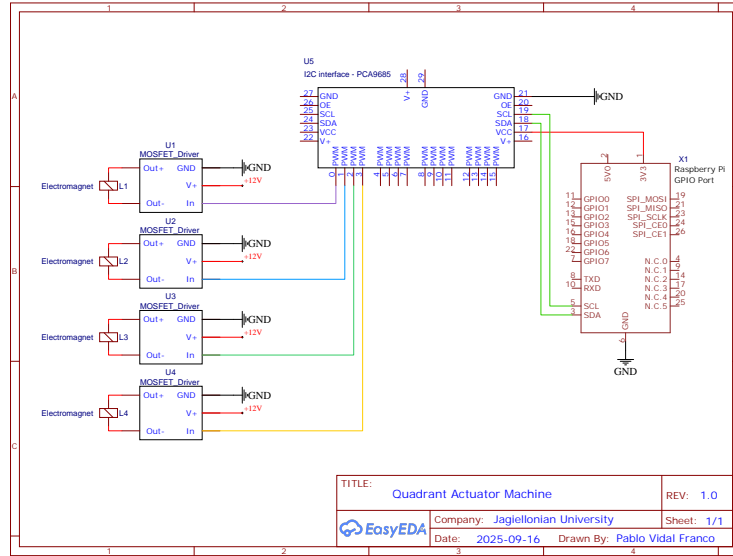


Figure 3: Connections of the QAM.

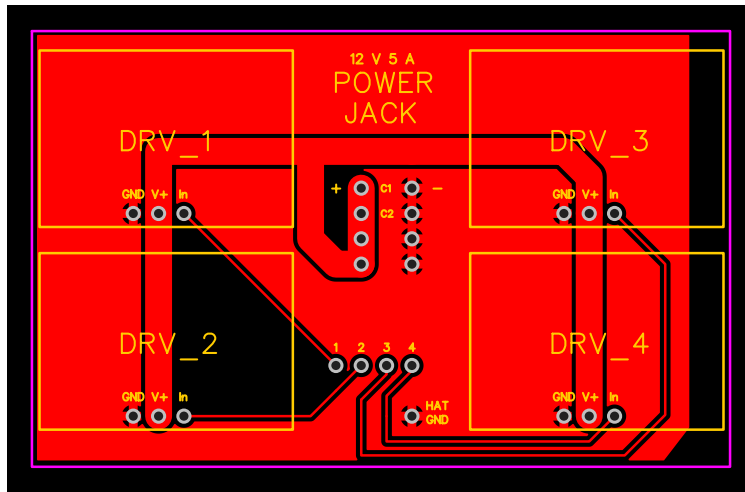


Figure 4: QAM PCB.

3 Construction

Figure 5 shows the design of the 3D-printed parts and how the Petri dish and solenoids fit within it.

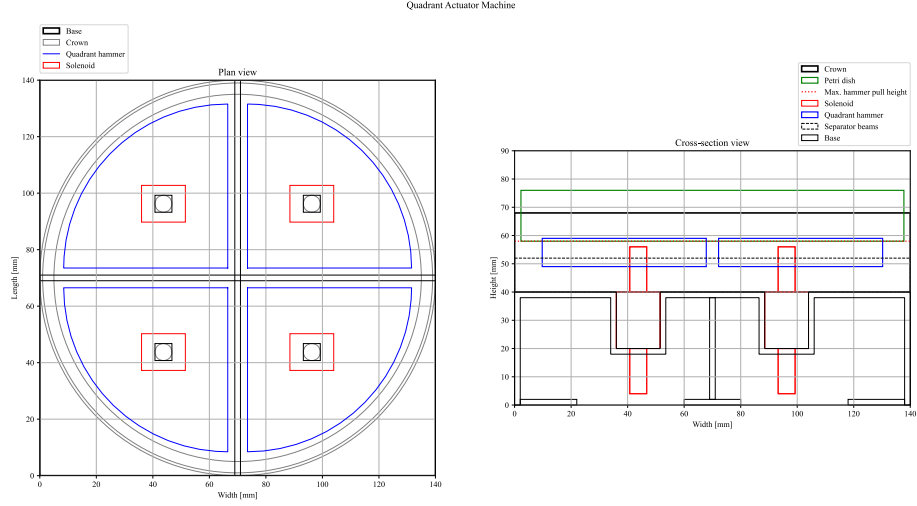


Figure 5: Plan view and side view of the Slim QAM.

4 Programming

The I²C PWM was programmed in Python to drive the solenoids as shown in Table 2. The quadrants are denoted by Q_i where i is any of the 4 available quadrants in the Petri dish and $Q_i \in \{\text{OFF}, \text{LOW}, \text{MEDIUM}, \text{HIGH}\}$.

Table 2: Values of Q_i and descriptions.

Value	Description
OFF	The electromagnet is not being acted upon.
LOW	The electromagnet taps the Petri dish 4 times per second.
MEDIUM	The electromagnet taps the Petri dish 8 times per second.
HIGH	The electromagnet taps the Petri dish 16 times per second.

The Python file `quadrant_operation.py` contains all the relevant code. To operate the PCA9685, the following library has to be installed: `adafruit-circuitpython-pca9685`. For example, to activate only the fourth quadrant at 4 taps/s, the script can be called like this:

```
pi@raspberrypi:~/qam$ quadrant_operation.py LOW [0,0,0,1]
```

5 Example dataset obtained using the QAM

The QAM was tested by performing experiments on a model organism to gather data to extend the Vicsek model (a swarm behavior model).

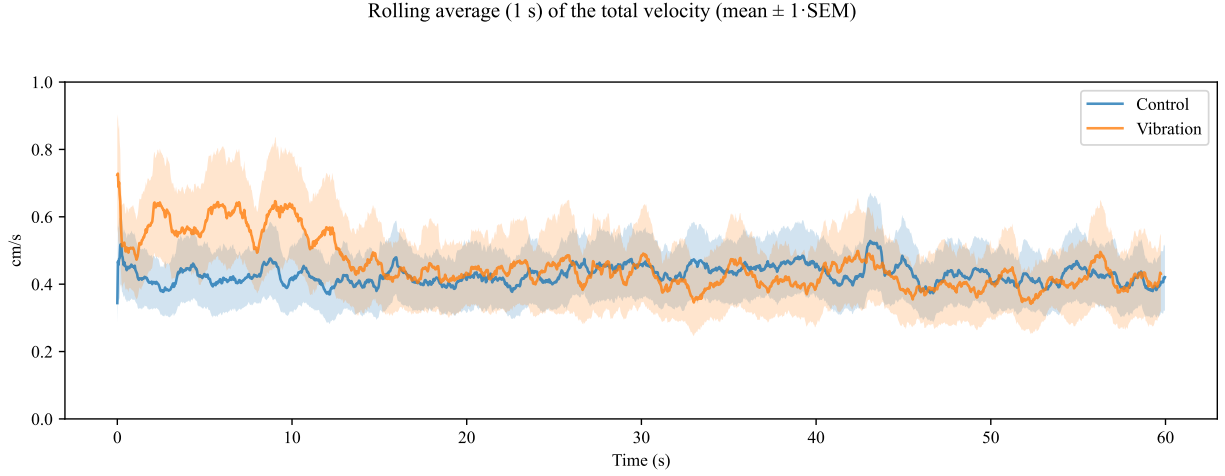


Figure 6: Control vs Vibration setup on the QAM (speed).

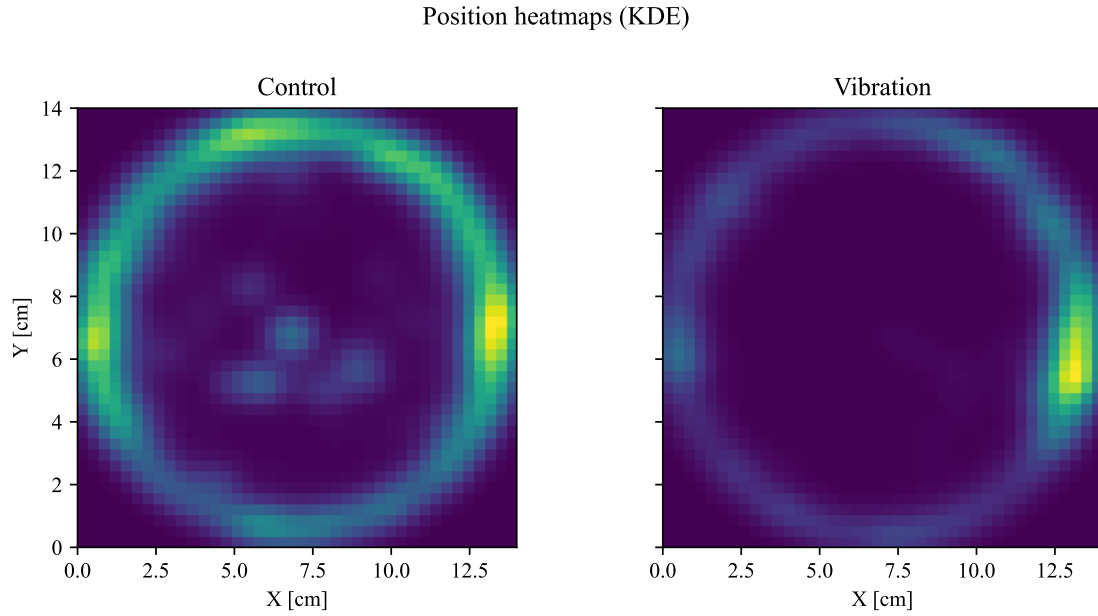


Figure 7: Control vs Vibration setup on the QAM (KDE heatmaps). Both leftmost quadrants were active in the Vibration group. Note the avoidance of the active quadrants.