CycBot: A Cyclical Soft Robot for Non-verbal Communication with Humans

Alan Bejnarowicz[1], Jan Górski[2] and Maciej Piszczek[3]

***Abstract*— In this project, we present CycBot — a soft robot designed for non-verbal communication with humans. The robot's design incorporates silicone materials and 3D printing. Our robot was inspired by toys like Furbies [1] and Tamagotchis [2], aiming to create an engaging interaction experience. We achieved a successful design and fabrication of CycBot, demonstrating its ability to express emotions using soft robotics actuator. In our Human-Robot Interaction (HRI) experiment, we evaluated the robot's effectiveness in conveying emotions and engaging users.**

I. Introduction

A. Design of the robot

Our robot, CycBot, was designed in Fusion 360 and fabricated using a combination of 3D printing and silicone molding. The robot's body consists of a 3D-printed frame, while the body and ears are made from Ecoflex 00-30 silicone. The robot uses one servo motor to rotate head, three air pumps, and two air valves to control the inflation and deflation of the ears. To make the robot more human-like, we cast 'belly' and ears using skin-coloured silicone. To keep the robot upright we poured small amount of hard silicone on the bottom of its belly. The robot is controlled using the Arduino Uno microcontroller [3], which manages the servo motor and air pumps/valves. The robot's behaviour is programmed to respond to human interactions, allowing it to express different emotions through ear movements and head rotations.

We designed our robot to show five distinct emotions: interested, happiness, annoyed and sadness. Each emotion is expressed through a combination of ear movements and head rotations, based on the human input. The inflatable belly of the robot has a pressure sensor that detects when a human touches it and for how long.

II. Methods

A. Fabrication

The hardest part of CycBot to make, was the silicone belly. We designed and 3D printed a mold shown in Figure 2. The mold consists of four parts: split outer shell, inner core and cap to hold the inner core in place. The first step in making the silicone belly was to assemble the mold by placing the inner core inside the outer shells, and securing it with the cap. Then, we sealed all edges with duct tape to prevent silicone leakage. Because of the wall thickness of the belly (3 mm on the walls and 8 mm on the bottom) and the mold being relatively tall, we were concerned about air bubbles getting trapped inside the silicone during pouring, which would make our part faulty. To keep our silicone bubble free, we put our silicone mixture in a vacuum chamber for 15 minutes and then poured it slowly into the mold through a small hole on the top of the mold cap. After pouring, we placed the mold in a vacuum pot for 45 minutes to further eliminate any remaining air bubbles.

[1]albej24@student.sdu.dk
[2]jagor24@student.sdu.dk
[3]mapis24@student.sdu.dk

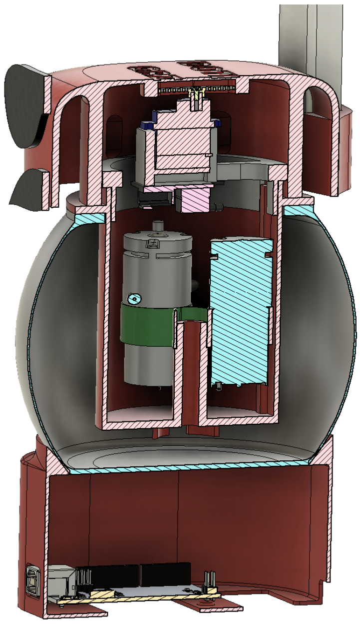


Fig. 1: CycBot section view showing internal components and silicone parts.

Fortunately, our part came out bubble-free on the first try. We then placed the mold in an oven at 60°C for 2 hours to cure the silicone. We put our mold in the oven for longer than recommended by the silicone manufacturer to make sure the silicone was fully cured, because 3D printed molds tend to absorb some heat during the curing process.

After curing, we disassembled the mold and removed the silicone belly. The final step was to trim any excess silicone and glue the belly to the robot frame using Sil-Poxy silicone adhesive. Figure 1 shows a section view of the CycBot with all internal components and silicone parts.

In the end our mold design worked very well, and we were able to make a high quality silicone belly for our robot.

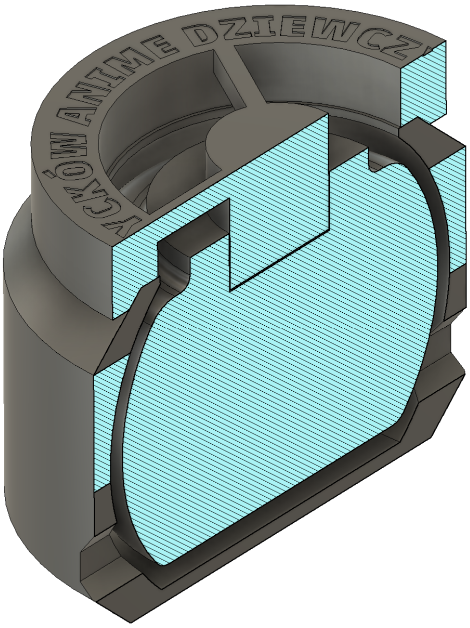


Fig. 2: 3D model of mold used to cast CycBot's silicone belly.

B. Hardware and software

The robot is controlled by the Arduino Uno microcontroller, equipped with a motor shield [4]. The motor shield has only 4 motor outputs, so it was a limiting factor for the design. That meant, there could be only 4 pumps/valves used for robot the actuation. The code used in this project can be found on GitHub.

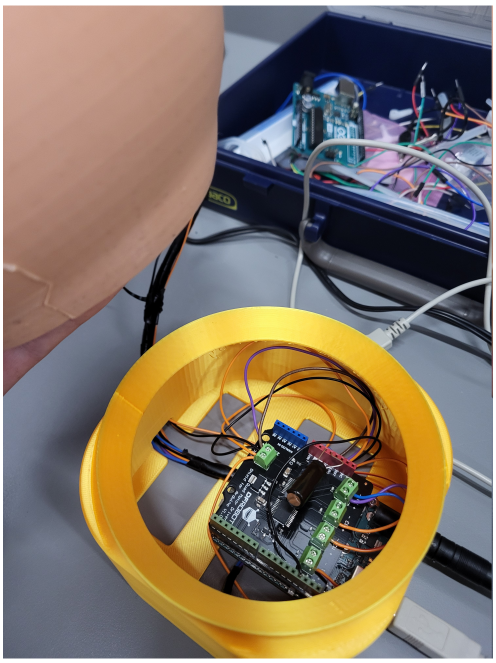


Fig. 3: Processing unit hidden inside the robot's base.

The final design incorporates 3 air pumps, 2 solenoid valves, 1 servo motor and 1 differential pressure sensor. The power source is a 5.9 V, 2 A power adapter, but, as it turned out not to be enough, the Arduino board has to be powered separately via USB cable. To overcome the limited number of motor outputs, the 2 solenoid valves share the same motor output of the motor shield. Because of that, the robot's ears could not be deflated independently.

The first pump, is responsible for the belly inflation. It is coupled in the feedback loop with the pressure sensor. Its readings are read by the Arduino's ADC and filtered using an alpha-beta digital filter [5]. Then, they are used by the PI controller to maintain the desired air pressure inside the belly.

Additionally, sudden changes in pressure readings can be detected, and counted as touches. Timers are used to distinguish between the short and long presses. After a touch is detected, the dead zone timer is started to prevent multiple touch detections from a single press.

The other two pump are responsible for inflating the robot's ears. The ears can be inflated independently, but as they are connected through the solenoid valves to the same motor shield output, they can only be deflated simultaneously. There are also no pressure sensor for the ears, so they have to be controlled by a feed-forward controller.

The servo motor is used to move the robot's head. It is powered directly from the power adapter to avoid voltage drops on the microcontroller.

C. Robot's behaviour

TABLE I: Robot expressed emotions.

Emotion	Ears movement	Head movement
Sad	Both ears down	—
Interested	—	Slow head swipe
Happy	Move both ears one at a time	—
Annoyed	—	Rapid head movements

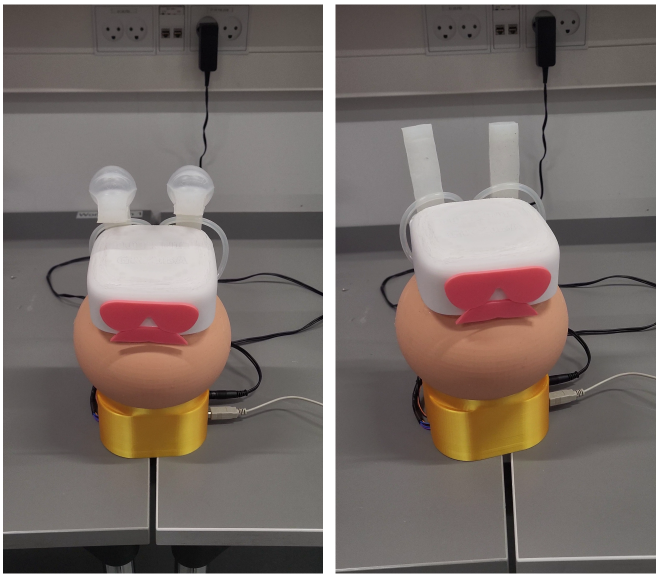
The robot operates based on a state machine that defines interactions with humans (see Table I and Table II). The robot starts in the default, sad state (see Figure 4a), where it waits for human interaction. Both ears are inflated to point downwards, indicating sadness. When a single short touch is detected, the robot transitions to the interested state (see Figure 4b), where it slowly swipes its head side to side, showing curiosity. If no interaction occurs for 20 seconds, the robot returns to the sad state.

A single short touch in the interested state makes the robot happy (see Figure 5a). In this state, the robot moves its ears one at a time, expressing joy. If no interaction occurs for 20 seconds, the robot goes back to the sad state. However, if a double or long touch is detected while happy, the robot becomes annoyed (see Figure 5b). In this state, it rapidly moves its head side to side, indicating irritation. After 10 seconds, the robot calms down and returns to the interested state.

The state machine design is also shown in Figure 6.

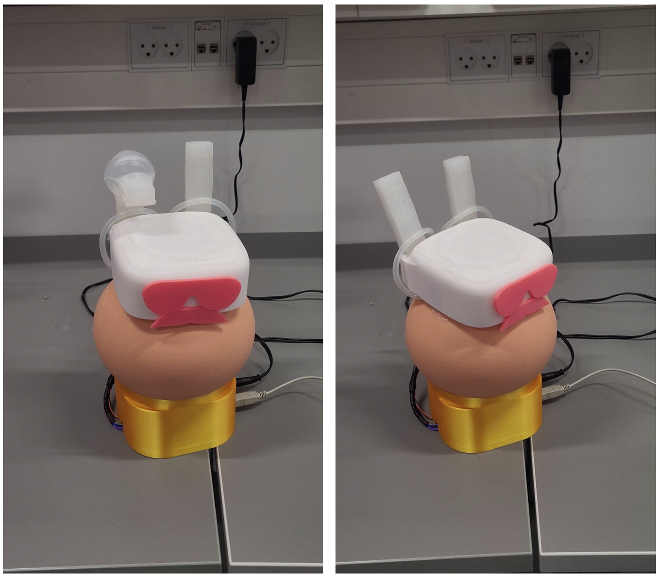
D. HRI experiment

Signals used in experiments are robot behaviours explained in Table I.



(a) Sad state. (b) Interested state.

Fig. 4: CycBot in different emotional states.



(a) Happy state. (b) Annoyed state.

Fig. 5: CycBot in different emotional states.

TABLE II: State changes.

State change	Cause
Sad → Interested	Single short touch
Interested → Sad	No interaction for 20 s
Interested → Happy	Single short touch
Happy → Sad	No interaction for 20 s
Happy → Annoyed	Double or long touch
Annoyed → Interested	After 10 s

Hypothesis H1: Participants will touch robot more while presenting Interested or Annoyed expression. **Measure:** Number of touches during each behaviour. **Analysis:** One-way ANOVA test to compare means of touches during each behaviour.

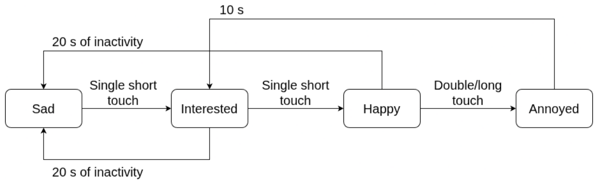


Fig. 6: CycBot state machine diagram.

III. Results

A. HRI experiment data

Participants interacted with the robot while it expressed different emotions in a cycle shown in Figure 6. Participants could touch the robot's head or belly freely during the interaction. We recorded number of touches during each robot behaviour. The collected data is summarized in Table III.

TABLE III: Count of touches during each robot behaviour in HRI experiment.

Expression	Touch count
Idle	—
Interested	—
Happy	—
Annoyed	—
Sad	—

IV. Discussion

V. Conclusion

Appendix

YouTube video

The video demonstration of this project can be found on YouTube.

Code

The code used in this project can be found on GitHub.

Survey

The survey used in the HRI experiment can be found on SurveyXact.

Contributions

1) Conceptual contributions (Alan X%, Jan X%, Maciej X%)
2) Writing/implementation of control code (Alan X%, Jan X%, Maciej X%)
3) Design and manufacturing (Alan X%, Jan X%, Maciej X%)
4) Production of experimental data (Alan X%, Jan X%, Maciej X%)
5) Analysis and presentation of experimental data (Alan X%, Jan X%, Maciej X%)
6) Writing report (Alan X%, Jan X%, Maciej X%)

REFERENCES

[1] *Furby*. [Online]. Available: https://shop.hasbro.com/en-us/furby (visited on 12/06/2025).

[2] *Tamagotchi*. [Online]. Available: https://tamagotchi-official.com/gb/ (visited on 12/06/2025).

[3] *Arduino uno*. [Online]. Available: https://store-usa.arduino.cc/products/arduino-uno-rev3 (visited on 12/06/2025).

[4] *Quad dc motor driver shield for arduino*. [Online]. Available: https://www.dfrobot.com/product-1513.html (visited on 12/06/2025).

[5] *Alpha beta filter*. [Online]. Available: https://etola710.github.io/filters/mathematics/alpha-beta-filter/ (visited on 12/06/2025).