TECHNICAL UNIVERSITY
OF CLUJ-NAPOCA
ROMANIA

FOOD DISPENSER FOR PETS

Fodor Zsófia

Computer Science

2020

Contents

1. Introduction

I chose to assemble and implement a food dispenser for pets, because having a dog, sometimes it is easier to automize some features, and allocate more time for other aspects, regarding my best friend. There are also times, when my dog wakes me up at night, because she wants to eat, so with this device I would not have to wake up, I would only have to connect to the food dispenser from my phone, and send it a command to pour a certain amount of food in the dog's bowl.

The scope is to create a dispenser which stores food, and by pushing a button or sending a command via Bluetooth from our phones will pour a certain amount of food for our dog/cat.

It is a widely used idea, but I personally never thought of using something like this. But now, seeing that it spares me some time, even if just a few minutes, I think it is a useful object to have in the house. There are several implemented ideas, and many different montages to choose from.

2. Bibliographic research

While searching for inspirational resources I found some videos of how these food dispensers work, which gave me an overall idea of how my project should look like after completion.

I also used the **www.arduino.cc** link for checking function signatures and usage examples.

The project that is most similar to mine, which I found on the internet, can be found on the following link: https://www.deviceplus.com/arduino/make-a-smart-automatic-pet-feeder-with-arduino-uno/

Perspectives	My Project	Already existing solution
Cost of resources	111 RON	149 RON
Adaptation to user's needs	User can pour the food either with the help of the buttons or from the phone, sending commands via Bluetooth module	User needs to program the time at which the food should be poured, and everything is done automatically. It measures the distance from the bowl, that is how it knows the size of the portion to pour. It also has a light sensor to distinguish day from night.
Usage of servo motor	Attached to food recipient, the bottom of the box was cut off and attached to the arm of the motor, which when activated moves the bottom of the box, opening it and letting the food drop to the bowl	A string is attached to the arm of the servo motor, which when activated pulls the string which in return moves the bottom of the recipient, opening it and pouring the food.

3. Solution and Implementation

3.1. Overall description

After I completed the montage, namely: mounting the buttons, leds, light sensor, Bluetooth module, buzzer and performed some tests, I started the real montage, i.e., making my food dispenser usable.

I used a Pringles box for the food recipient, I cut off the bottom of the box, which I later attached to the arm of the servo motor. Then I glued the servo motor on the box.

I also made a stand to which I attached this recipient, so that I do not have to hold it every time I test it or use it. I attached the breadboard and the Arduino mega board to this stand also.

An overall description of how it works:

When started, the recipient has to contain some food, not too much, because the arm of the servo motor can not hold too much weight. We can choose to either manipulate the servo motor by pressing the button corresponding to our desire, or by Bluetooth via our phone. If we choose the second option, it is advised to be close to the Bluetooth module, for proper functioning, because sometimes the connection can be lost. On our phones we have to open an application, specifically designed for Bluetooth communication, for example Bluetooth Terminal. Here we should get a message of the possible commands we can send to our board. If not, then one should know that there are four commands, for four different amounts of food: small portion, medium portion, large and extra-large portion. In the Bluetooth terminal we have to press the buttons a,b,c or d for the above commands.

After the command was sent via Bluetooth or a button was pressed, the servo motor will be activated and it will move the bottom of our recipient, letting the food fall out of it. When the instruction is executed, the buzzer will sing the specific sound for each amount, informing us, that the food was poured. If the leds are on, informing us, that it is dark, then the buzzer will only play a short tone, not an entire song.

In parallel the photoresistor is always functioning and will turn on the leds if it is dark. So that we can see the buttons, or the bowl even if it is dark.

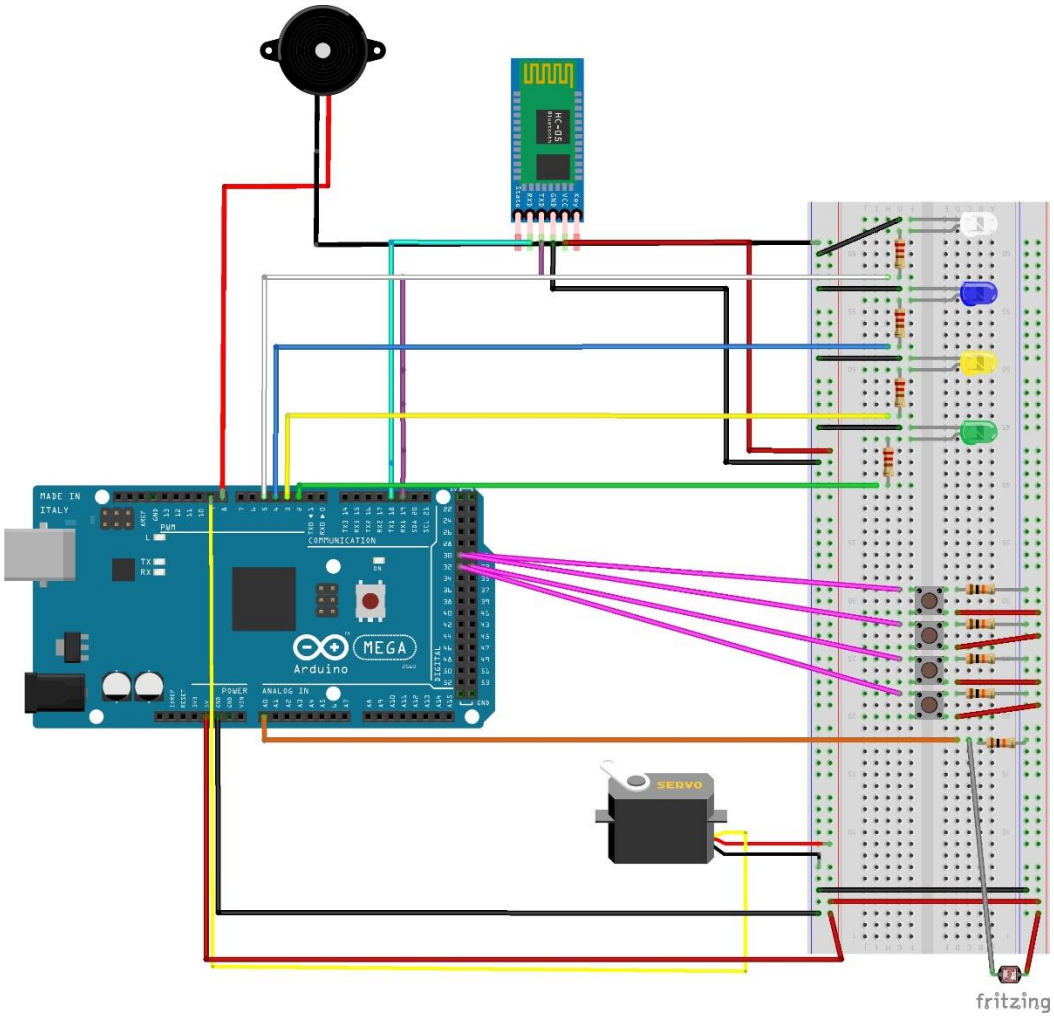
3.2. Theoretical description

We need to manipulate the servo motor in a way, that it can pour different amounts of food, depending on the instruction we execute. We have to write a song or just a sound for our buzzer to signal us that the food was poured in a bowl. And lastly, we need to use the photoresistor to turn on the leds if it is dark outside. The value read from the photoresistor should be inversely proportional to the intensity of the leds.

3.3. Implementation

- Hardware
  The project consists of the following components:
  - 4 buttons
  - 4 leds (white, blue, yellow, green)
  - Bluetooth module

- Servo motor
- Buzzer
- Light sensor
- Arduino mega board



- Software

  There are no complicated functions, I created four functions, for the four different amounts of food, which only differ in the amount of time the servo motor waits before it moves back to its initial state.

  For the servo motor, I used the servo.write(x), where x specifies the angle the arm of the motor should write.

  For the photoresistor, I mapped the value, which was read from the photoresistor, via de analog pin, to the [0,255] interval. As I said above these two values (the value red from the photoresistor and the intensity of the led) are inversely proportional. So, I mapped the value from the interval [0,1023] to the interval [255,0].

  The Bluetooth module was connected via serial communication, i.e., the RX pin from the module was connected to the TX1 pin on the board, and the TX pin from the

module was connected to the RX pin on the board. For writing on the Bluetooth terminal, I used the Serial.write() function.
For the buzzer I wrote four different melodies, for the four different amounts of food.

4. Testing and validation

At the beginning I wanted to connect a 4-button component instead of connecting four buttons separately. The problem was that this component had to be connected to an analog pin, and the values that were read were ambiguous. Meaning that even if I touched the component, without pressing any of the buttons, it read that one, or even two buttons were pressed. Unfortunately, I could not solve it to work properly, so I chose to implement four buttons separately, which could be connected to digital pins, and this way the values were read correctly.

When implementing the servo motor, I had to think about the angle the motor should move, because I needed it to move only into one direction, and back the same amount. The servo motor is initially in 0, which means that it can only move 90 degrees to the left or 90 degrees to the right. And only after it moved 90 degrees in one direction, it can move 90 degrees in the other direction. I was not sure if those 90 degrees would be enough to entirely open the recipient, but it proved to be enough. So, the servo motor moves to 90 degrees, then it comes back to 0, its initial state.

I wanted the leds to only light up, when it is dark outside, or at least when it is dusk. That meant that I only had to activate the led, when the read value on the photoresistor was small, around 2-300. I put the values in a table so we it is clearer what I tried to solve

Read value interval -> led intensity	When is led turned on?
[0,1023]-> [255, 0]	Even if it is not dark outside, the leds are on. But we do not need them.
[0, 700] -> [255, 0]	The above statement is still available.
[0, 300] -> [255, 0]	The leds turn on, even though the light in the room is turned on. In that cease we do not need them to be turned on, only if the visibility is reduced.
[0, 200] -> [255, 0]	This seems to be the proper interval in which we should turn on the leds.

The photoresistor was placed on the top of the stand, where the light source is the strongest and no other component disturbs it.

At the beginning I assembled the buttons and tested them. The I added the buzzer, which signaled if a button was pushed. After adding the buzzer, I added the leds together with the photoresistor, and tested how the values should be mapped according to the value read from the photoresistor. Then I added the servo motor and tested if it works first without pressing the button, then I tested if it works when pressing a button. Lastly, I added the Bluetooth module,

which basically did the same thing as the buttons before, only that we did not press a button, we sent a signal to the module, telling it what action the board should execute.

5. Conclusion

The purpose of this project was fulfilled, since the idea was to remotely feed the dog, or if not remotely, then only by pressing a button, not having to pour the food by self.

More tests were run on this project, for it to function accurately. Firstly, I had to adapt the photoresistor to my needs, I wanted the leds to be turned on, only when the visibility is reduced, otherwise it is not needed. Another aspect was the servo motor, and attaching the bottom of the dispenser to it, since the arm of the servo motor is not strong enough, it can not hold too much food.

This project is helpful for lazy people, who do not like to do thing by themselves, they would rather do everything from the comfort of their couch or bed. This food dispenser can be used from a small distance, but if the Bluetooth module is replaced with a stronger one, it can also be controlled from the couch, or other rooms. By just pressing a button, one can feed its pet.

Some improvements that can be made in the future:

- Attaching a stronger arm to the servo motor, so that it can hold a larger amount of food
- One could attach an ultrasonic sensor, for the dispenser to pour food, when the pet is approaching its food bowl.
- One could find another idea of measuring the amount of food, by using a sensor for example.