*Microprocessor Systems*

**Smart Feeding Device for Pets**

Amelia Munteanu, Maria Teodor

**Content:**

1. User Requirements

1. System overview
2. Hardware design
3. Software design
4. Python modules
5. Repository
6. Results and further work
7. References
8. **User requirements**

The Smart Feeding device must succeed in dispensing food at user requests or automatically, based on a schedule and an ultrasonic sensor. The system must provide the proper amount of food every time and it should also be permanently accessible. The access to the system should be provided via an easy-to-use Web interface.

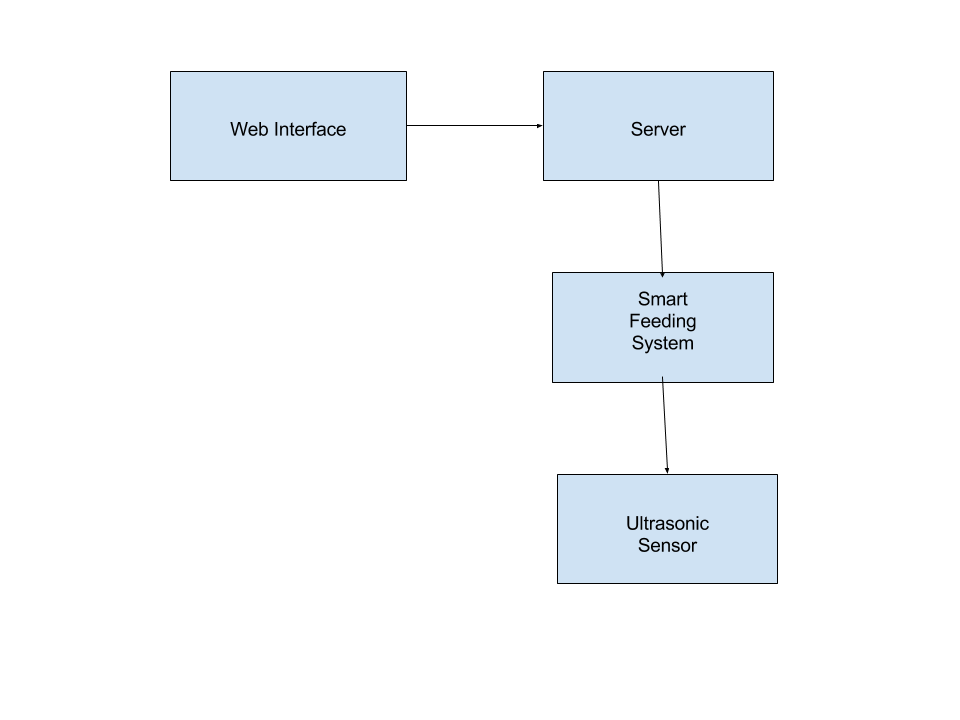
1. **System Overview**

Figure 1: System Overview

The server, created using the Flask web service, has the purpose of controlling the Smart Feeding System by rotating the stepper motor attached to a portioned food container (**Figure 2**) such that it performs 1/8 from a full rotation whenever it receives a request from the client or when the dispenser is automatically triggered.

The client can send a request to the server by accessing a page within the Web browser.

The ultrasonic sensor is an extra feature of the Smart Feeding Device, which enables the option of automatic feeding, the mechanism of dropping food being triggered whenever the sensor detects an obstacle within a previously established distance range and after a certain period of time has passed since the previous dropping.

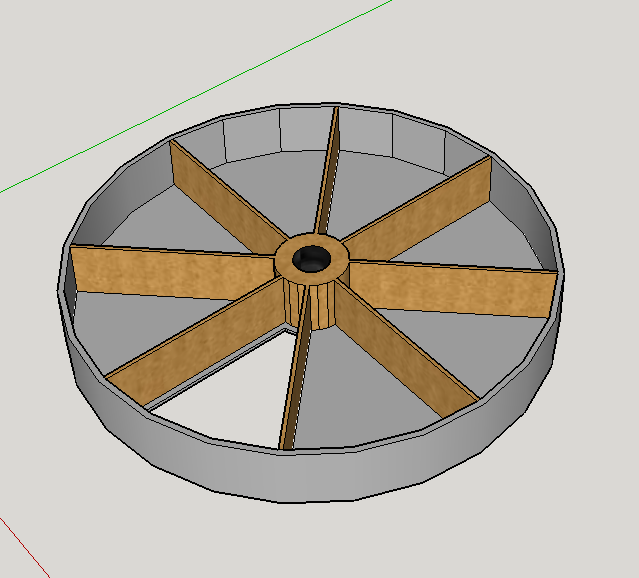


Figure 2: Portioned food container

1. **Hardware design**

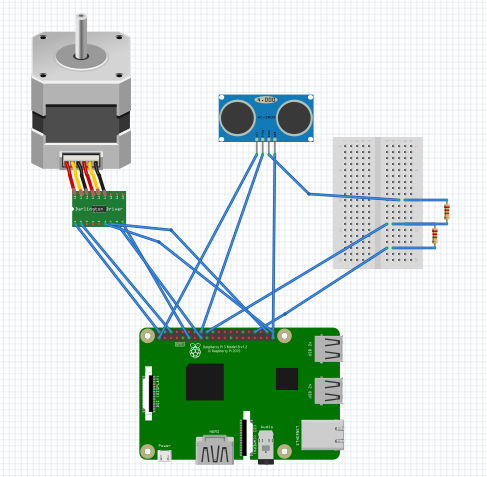
****

Figure 3: Hardware Diagram

We implemented the application on a Raspberry Pi 3 because of its versatility, ease of use and wide range of capabilities. One of its critical features used in our application was its Wi-Fi connection, used to host the web server.

As it can be observed in **Figure 3** a **28BYJ-48 Unipolar Stepper Motor** is used together with a **ULN2003 driver board**. The motor connects to the controller board with a pre-supplied connector. There are 6 pins that have to be connected to the Pi header (P1):

* Inp1 (P1-11)
* Inp2 (P1-15)
* Inp3 (P1-38)
* Inp4 (P1-40)
* 5V (P1-02)
* GN D(P1-06)

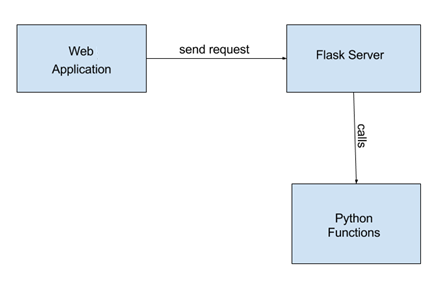
The input voltage for the motor is 5V so we can use the Raspberry Pi as the power supply. The stepper motor is used to rotate the portioned container in order to drop a precise amount of food. We do this by providing a sequence of ‘high’ and ‘low’ levels to each of the 4 inputs using half stepping mode.

Half Stepping is a combination of single and dual phase stepping, alternating between one and two coils being turned on. In half stepping one cycle takes 8 steps compared to the full stepping which takes 4 steps, thus doubling the resolution of the step motor resulting in smoother rotations. Our container is divided in 8 partitions but only one partition is rotated at a time so in order to achieve that a range of 64 cycles is used.

1 revolution = 8 cycles; gear ratio= 1/64 => 1 revolution = 512 cycles

1/8(meaning 1 partition) rotation => 512/8 = 64 cycles.

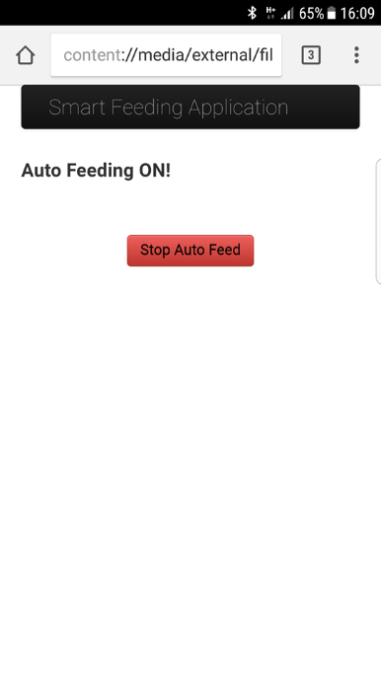
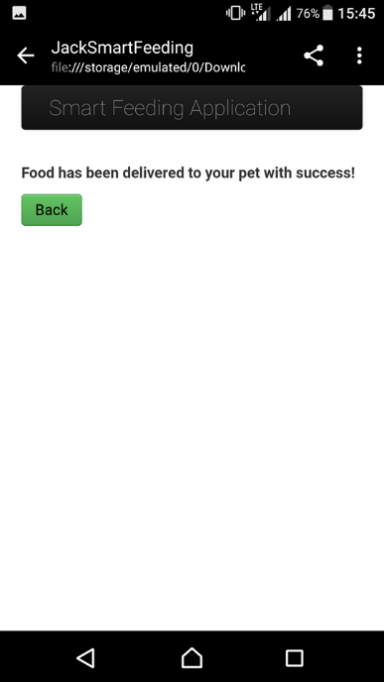
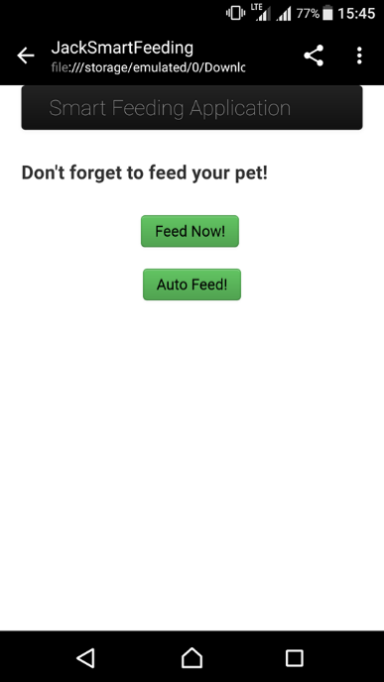
In our circuit a HC-SR04 Ultrasonic sensor is used. Connected to the Raspberry Pi through the breadboard, it uses a 5V (P1-04) Vcc, GND (P1-09), Trig (P1-16), Echo (P1-18). This sensor is used when the automatic feeding mode is on and the pet is in a certain distance range from the feeder. After a food drop is performed, a timer is started and for a set amount of time the feeder will not be activated no matter what the sensor dictates.

1. **Software design**

The web interface consists of one html page. Index.html is the page being displayed in the browser whenever a client navigates to the address of the Raspberry PI board, followed by the port on which our server is listening. It has two buttons, corresponding to the two request types a client can send to the server: deliver food to his pet with one simple click (“Feed Now!” button) or enable automatic food delivery whenever the pet gets close to the sensor (“Auto Feed!” button).

When the “Feed Now!” button is clicked, a message saying that the food was successfully delivered is displayed. While the “Auto Feed!” button is clicked, it is disabled so it cannot be clicked again and a “Stop Auto Feed” button appears, so it can be clicked when need to end the auto-feeding process.

Screen captures with the 3 html pages can be seen below:



When the “Feed Now!” button is pressed, the python function feed() is called, because the @app.route decorator was used in order to match the redirect URL with our function.

The function feed() simply starts the stepper motor, which will make 1/8 of a full rotation, thus the container will drop one portion of food.

When the “Auto Feed!” button is pressed, the python function autoFeed() is called in a similar manner, but this time it will spawn another process that will run a routine, waiting for the sensor to detect the presence of the pet near the device. If the pet approaches the front of the device, closer than 20cm, the function feed() will be called, dropping another portion of food. This will only happen if at least 4 hours have passed since the last food drop.

For creating the server we used Flask, a micro-framework which provides tools, libraries and technologies for building web-based applications.

1. **Repository**

The project schematics, diagrams and codebase are located on the following git repository:

**https://github.com/MariaTeodor18/Travian**

1. **Results and further work**

The submitted version of the project offers the following functionalities:

* Dispensing food to your pet remotely, with a single button click
* Enabling auto feeding, a feature which will feed your dog whenever it is near the device (between two food dispenses, a time period of at least 4 hours should pass)

For the next iteration, the following improvements are planned:

* Improving the web interface for the client
* Making our local server accessible over the internet

1. **References**
2. Project Repository, Web:
3. Google Drawings, Web: https://docs.google.com/drawings/create/
4. Fritzing, Web: http://fritzing.org/
5. Flask, Web: http://flask.pocoo.org/
6. 28BYJ-48 Stepper Motor Datasheet, Web: http://robocraft.ru/files/datasheet/28BYJ-48.pdf
7. HC-SR04 Ultrasonic Sensor Datasheet, Web: http://www.micropik.com/PDF/HCSR04.pdf