Autonomous Pet Food and Water Dispenser

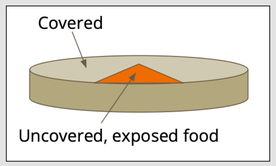
Tiger Li, Stephanie Bai, Panda Atipunumphai, Laura Bryant
Cyber-Physical Systems: Foundations and Project

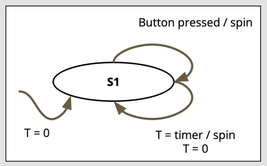
Summary

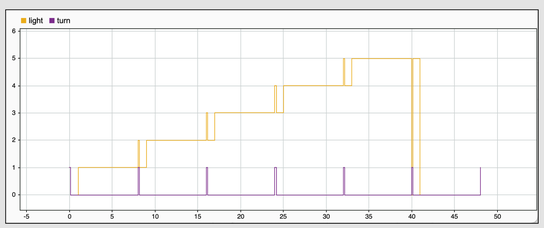
We made an automated water and food dispenser for a pet cat or dog. This would be useful for taking care of a pet, especially if the owner is away as the system will be able to function on its own. The produce is fully autonomous and able to regulate the food and water completely on its own without supervision for an extended amount of time. The machine is be connected to a valve that controls the flow of water into the bowl, as well as a valve that controls the flow of water out. It is able to detect when the water level is below a certain threshold and fill the water bowl up again. There is also be a timer that refreshes the water in the bowl after a certain amount of time has passed in order to ensure that the water that the pet has is always fresh and clean. We hope that this product will make it easier for owners to take care of their pets, and reduce the amount of times that a pet is left without water because an owner is forgetful or busy.

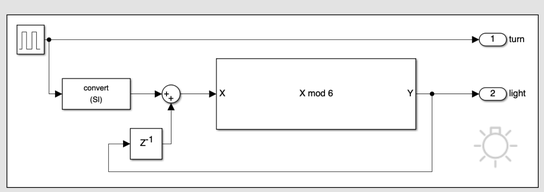
Food Bowl Model

The food supply delivers a fixed amount of food at a fixed time throughout the day specified by the owner. The amount of food dispensed is measured using a load cell, and the food is dispensed using a motor that opens a valve to the food supply and closes it when food is filled to the desired amount. Both the water and food supply are placed above the bowl such that a valve opening and closing will fill the bowl without needing pumps.







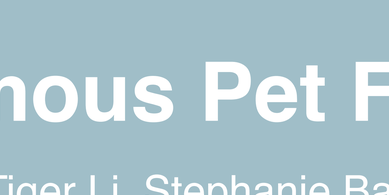


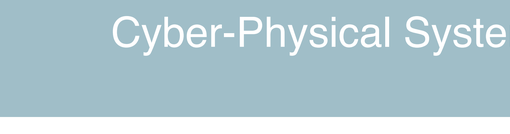
Constraints & Challenges

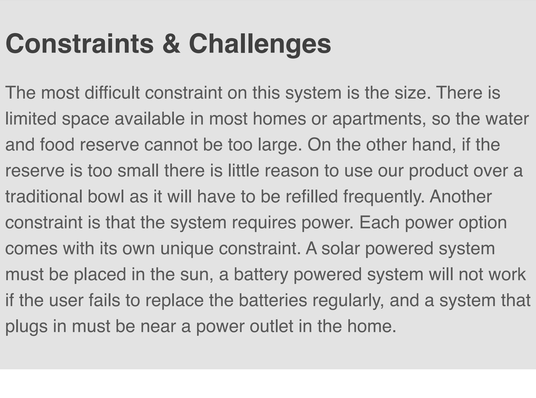
The most difficult constraint on this system is the size. There is limited space available in most homes or apartments, so the water and food reserve cannot be too large. On the other hand, if the reserve is too small there is little reason to use our product over a traditional bowl as it will have to be refilled frequently. Another constraint is that the system requires power. Each power option comes with its own unique constraint. A solar powered system must be placed in the sun, a battery powered system will not work if the user fails to replace the batteries regularly, and a system that plugs in must be near a power outlet in the home.

Water Bowl Model

The system refills the water when the water is below a certain level. The water level is detected using a touch sensor, and the water will refill using a small motor that opens and closes a valve to the water supply. The water bowl is emptied out once at the end of each day and refills the bowl with clean water.

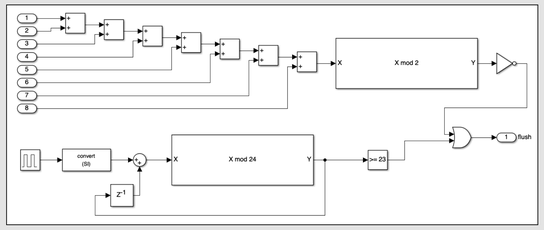




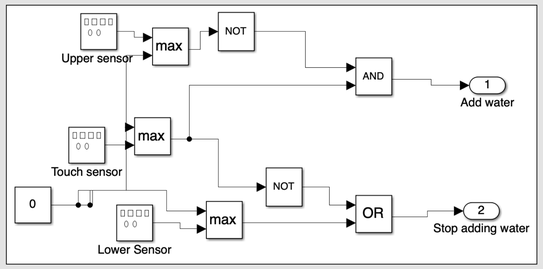


Complete System

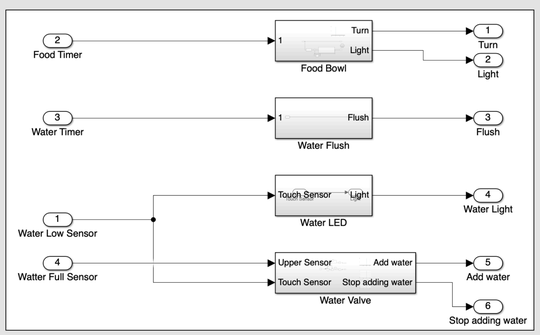
The system triggers lights to turn on when the supply is low and when it is gone so that the pet owner knows that it is time to replenish the supply. The system should allow the pet owner to leave the house and not have to worry about food and water for their pet for the whole day.



Assuming that the pet drinks approximately half of the bowl when it is thirsty, this model triggers the system to refill after the pet drinks twice. These inputs connect to the eight times that the pet becomes thirsty throughout the day and every other time the pet drinks, the system flushes out the water.

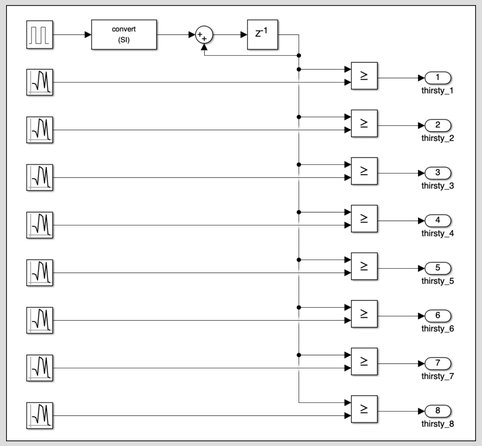


Since it is okay if the pet is hungry and it is more healthy that the pet be fed three times a day on schedule regardless of its hunger level, the food bowl's movement should not be dependent on the pet. The food bowl turns every eight hours independently of whether or not the pet is hungry, so it does not take input from the controller.



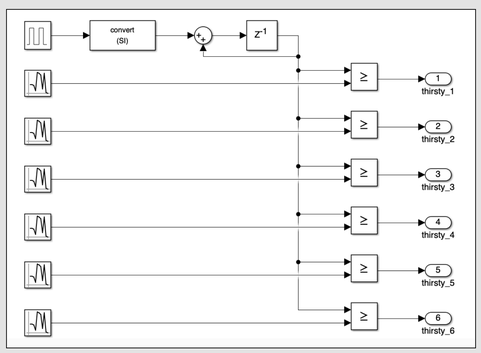
Pet Drinking

The pet drinks roughly every 6 hours, with a variance of one hour. This model has the pet drinking at random intervals within the appropriate range over a 48 hour period.



Pet Eating

The pet's eating habits are modeled by six random number generators that produce values within a reasonable range so that the pet gets hungry every 8 hours, with a variance of 2 hours.



Acknowledgements

Professor Gabor Karsai
Vanderbilt Department of Computer Science
Immersion Vanderbilt