



Memoria del proyecto

COMEDERO INTELIGENTE PARA LA GANADERÍA

CONTROLADO DE MANERA REMOTA

Asier Vega | Emma Murga | Gorka Berganza | Sistemas embebidos | 08/01/2022

Comedero inteligente remoto

Abstract—Nuestro proyecto se centra en ayudar a un ganadero a realizar sus labores diarias. Se propone la implementación de comederos automáticos, que puedan ser supervisados de manera remota a través de MQTT y que permitan una visualización de los datos a través de una aplicación web.

Palabras clave—IoT, Agroindustria, sistema embebido, automatización.

I. INTRODUCCIÓN

Este proyecto busca solventar el problema que tienen los ganaderos a la hora de alimentar a su ganado en épocas en las que el acceso al emplazamiento de estos es tedioso o poco práctico. Para ello incorpora un sistema de suministro de agua y alimento automático que reduce los viajes necesarios. De manera que el ganadero sólo tendría que ir cuando el nivel de los tanques sea muy reducido.

El sistema incorpora un servo para el suministro y sensores que detectan el nivel de los tanques, como suele ser común que el lugar donde se encuentra el ganado esté considerablemente lejos de donde reside el ganadero, existe una comunicación entre la RPi que controla estos sensores y actuadores encargados de la gestión de alimentos, y otra que está en el lugar del ganadero, la cual tendrá un led que indique cuando uno de estos sensores haya detectado un nivel demasiado reducido.

Esta idea surge del problema que se suele dar en invierno, cuando los animales no pueden pastar, y tienen que pasar mucho tiempo en los graneros. Tener que ir todos los días a alimentar a los animales, es una tarea que puede ser tediosa y más cuando las condiciones climáticas son adversas, es por ello que, mediante esta idea, se podrían reducir los viajes necesarios, y realizar solo los indispensables.

II. ESTADO DEL ARTE

A. Estado de la tecnología en la actualidad:

Actualmente, el uso de los comederos automáticos está muy extendido y normalizado por todos lados, no solo en el ganado de animales grandes si no en los pequeños también e incluso en animales domésticos pero la solución que proponemos va algo más allá, no solo se encarga de alimentar automáticamente a los animales, si no minimizar la cantidad de viajes que hay que realizar para rellenar los tanques, y mostrar un sistema de alarmas mediante el cual saber el estado del tanque sin la necesidad de ir hasta el emplazamiento.

B. Ejemplos Reales de Nuestra Idea:

A día de hoy, la empresa GEA, implementa este tipo de sistemas mediante su producto DairyFeed C 8000, este es un sistema que incorpora lo que vendría siendo nuestra idea, pero de una manera más elaborada, ya que calcula la cantidad de alimento necesaria para cada animal

dependiendo del peso del mismo y lo divide en 12 comidas diarias,

III. IDENTIFICACIÓN DE REQUISITOS

A la hora de realizar este proyecto hemos tenido en cuenta requisitos tanto por parte del hardware como del software como funcionales. Mediante la relación de todos ellos, hemos conseguido implementar una solución a nuestro problema, evitar en la medida de lo posible que un ganadero tenga que desplazarse todos los días a echar o llevar comida a el ganado.

Para lograr este objetivo, decidimos medir las distancias por el interior de dos tanques, sus niveles. El tanque de comida, lo medimos mediante un sensor de ultrasonidos, de manera que marcamos un nivel, y si el sensor detecta que hay más de ese nivel, consideramos que sigue habiendo comida suficiente dentro del tanque. Una vez detectada comida suficiente procedemos al rellenado del comedero, volvemos a comprobar la distancia y si se mantiene por encima del nivel preestablecido, la alarma se activará, porque se considera que ya no hay comida suficiente en el tanque. Es decir, todo dependerá de la distancia que marque el sensor en el momento que se mida. Si desde la primera medida no se sobrepasa el nivel, no se rellenará el comedero. Y si en la segunda medida tampoco se sobrepasa, no se activaría la alarma. También se puede dar el caso de que para la primera medida haya suficiente pienso en el tanque, pero para la segunda medida ya no se sobrepase el nivel, que sería lo habitual, que se rellene el comedero y aun así siga habiendo pienso en el tanque. Para el caso del tanque del agua ocurriría lo mismo, pero al revés, es decir, el nivel en vez de ser superado tendría que ser menor al nivel establecido. Y el rellenado se realizará para el bebedero. Por último, al final del tanque de pienso tendremos un servo que dejará pasar el pienso cuando vayamos a rellenar el comedero. Y en el tanque de agua, tendremos un relé que dejará pasar el agua (imaginamos: mediante un accionador y una tabla) cuando se vaya a rellenar el bebedero.

A. Identificación de requisitos hardware

Para poder controlar los sensores de entrada y que estos activen las salidas correspondientes debemos hacer uso de dos ordenadores y dos RPi.

En el caso de los ordenadores no son necesarios ordenadores muy potentes porque disponemos de una serie limitada de entradas y salidas. Además, no hacemos uso de todas las que disponemos. Aunque los ordenadores si deben ser lo suficientemente potentes para gestionar todas estas entradas y salidas de forma física, sino nuestro proyecto carecería de sentido real.

En el caso de las RPi sería necesario que cada una tuviera una microSD con memoria suficiente. Además, del espacio necesario para introducir las entradas y salidas respectivamente. También, una fuente de alimentación para cada uno de las RPis, siempre las podemos conectar al ordenador, para mayor comodidad.

Los dispositivos irán conectados de tal forma:

Un ordenador y una raspberry pi estarán conectados de forma alámbrica donde el ganadero solicite, pudiendo visualizar las alarmas (dos salidas), los dos leds, uno del comedero y otro del bebedero. Y pudiendo acceder a la aplicación web para visualizar los niveles.

Y el otro ordenador y la otra raspberry pi también estarán conectados de forma alámbrica dentro de la ganadería en la zona de los comederos y bebederos. La conexión entre la ganadería y el ganadero se realizará de forma inalámbrica. En esta RPi se encontrarán nuestros dos sensores, el servomotor y el relé. Además de ser la RPi que ejecute el programa.

Dividiendo el equipo reduciremos el consumo energético, intentando conseguir una vida útil más duradera.

B. Identificación de requisitos software

- Organización

La correcta organización del programa es un requisito esencial. Lo hemos dividido en la parte publicadora (ganadería) y la subscriptora (ganadero).

La parte publicadora la hemos dividido de tal forma: con un archivo principal donde llamamos al resto de archivos de esta parte, que son los funcionamientos de los sensores.

La parte subscriptora la hemos dividido de tal forma: el CSS, los dos HTML y el archivo que recoge la salida del led, además de realizar la conexión con el servidor.

Con esta organización si se desean realizar cambios, se puede acceder de una manera fácil y rápida a cualquiera de las dos partes y a su comprensión.

- Comentarios

Hemos realizado todos los comentarios que hemos visto oportunos, aunque algunos sean obvios, para su posterior visualización por parte de un no integrante del grupo.

Por ejemplo, un requisito puede ser usar un protocolo de comunicación que permita al sistema obtener lecturas por parte de los sensores con velocidades de lectura de menos de 1 segundo (es decir, como muy lento, 1 lectura por segundo). Después la implementación sería usar el protocolo MQTT.

C. Identificación de requisitos funcionales

En definitiva, el sistema embebido deberá realizar las siguientes tareas:

- Medir la distancia y la temperatura/humedad x2
- Activar, mover y parar servomotor
- Activar y desactivar relé
- Avisar de las alarmas con los leds (tanques vacíos)
- Visualizar desde la aplicación web las medidas de los tanques

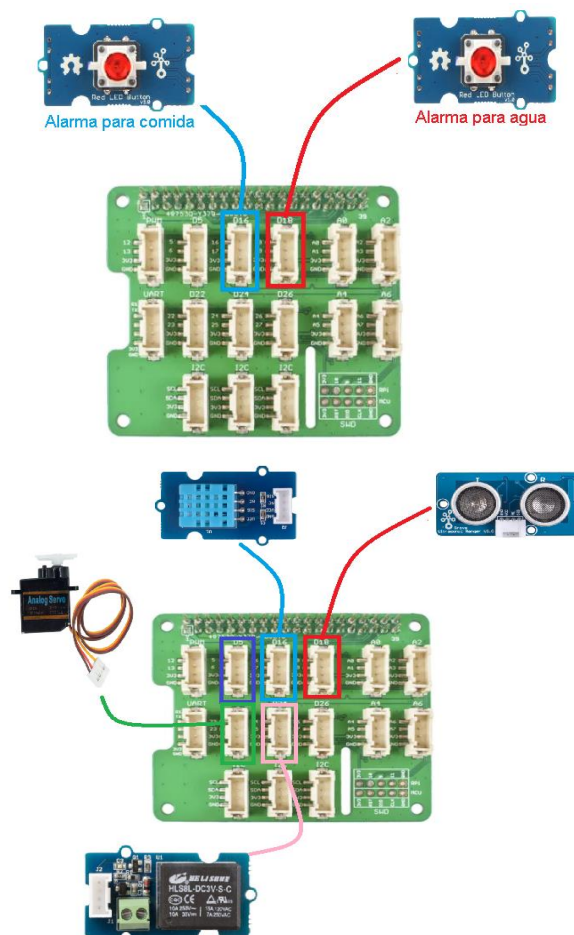
IV. IMPLEMENTACIÓN DEL PROYECTO

A. Descripción de componentes

A continuación se describen todos los components utilizados:

- Sensor de humedad, entrada digital DHT11 sensor de temperatura y humedad
- Sensor de ultrasonidos, entrada digital, para medir distancias aplicando técnicas de ultrasonidos.
- Servomotor, salida digital, controlado a través de un pin PWM a frecuencia 50.
- Relé, salida digital, para controlar la corriente alterna de un motor eléctrico.
- Led, salida digital, para indicar estados del sistema.

Esquema de conexiones:



B. Descripción software

Los archivos utilizados son los siguientes:

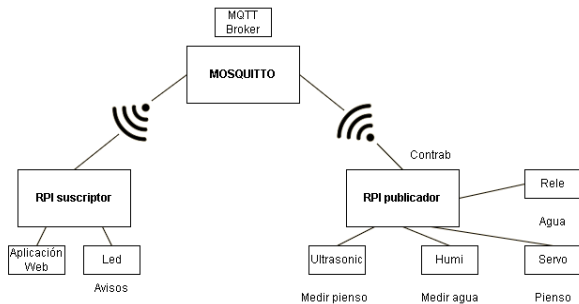
- Carpeta RPI subscriptora: “static” cuenta con un archivo css para una mejor visualización de la página html. “templates” cuenta con las dos templates que utiliza la aplicación web, un menú y una página para mostrar los datos. “AppServidor.py” archivo que contiene la aplicación servidora que recibe los mensajes de la RPi publicadora.
- Carpeta RPi publicadora, “dhtConfig.py”, es el archivo para leer los valores del sensor de humedad. “measureWidth.py” es el archivo para leer valores del sensor de ultrasonidos, “AppComedero.py” es el archivo que controla el comedero y envía los mensajes al broker.

Identify applicable funding agency here. If none, delete this text box.

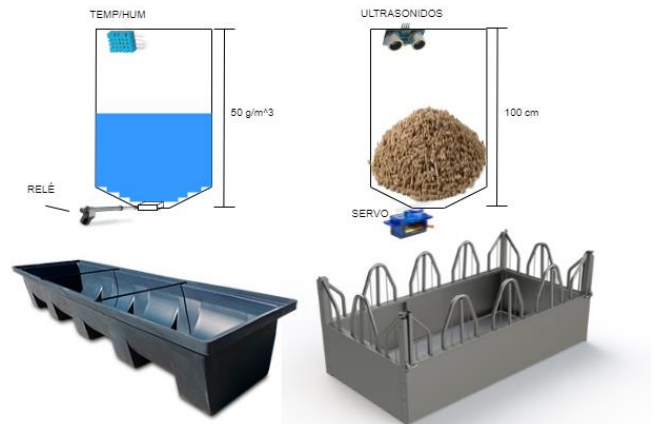
C. Descripción funcional del proyecto

Son dos partes, se utilizarán dos RPi. En la primera RPi, el suscriptor encontraremos el funcionamiento de la aplicación web, así como una salida. En la segunda RPi, encontraremos la parte más visible el publicador, donde tenemos un sensor de ultrasonidos que acciona un servo y un sensor de temperatura y humedad que acciona un relé. Sin tener en cuenta la nube bróker MQTT que es el servidor test.mosquitto.org.

Hemos incluido aquí un diagrama/dibujo del sistema implementado.



- RPi suscriptora, esta se suscribe a todos los topics MQTT que se publican desde las RPi publicadora, para ello se emplea un topic genérico (pub/comedero/#). Para mostrar los datos recibidos (nivel de humedad, cantidad de comida restante y avisos), se utiliza una aplicación web creada con el microframework Flask, la cual muestra los valores recibidos en una sencilla página HTML. La aplicación cuenta con una página de menú para acceder a múltiples RPi publicadoras (organizadas por un id) y otra página que muestra los valores para la RPi seleccionada. Por otro lado, el mensaje recibido cuenta con un campo de aviso. En el caso de producirse un aviso (falta comida o agua) se enciende el led indicando el aviso y se actualiza en cuando se recibe el mensaje.
- RPi publicadora, es la encargada de gestionar un comedero, que cuenta con todos los sensores y actuadores del sistema. Los mensajes recibidos del sensor de ultrasonidos y del sensor de humedad, se decodifican para poder ser entendidos. Después, se comprueba si es necesario rellenar o el comedero o el tanque de agua. En el caso que necesitemos rellenar el agua activamos un relé, que activa una bomba, durante un corto periodo de tiempo. Y en el caso que necesitemos rellenar la comida movemos un servomotor a determinados grados (cambiamos el duty cycle a determinados ms), durante un determinado tiempo y lo devolvemos a la posición inicial. Para cualquiera de los casos, se vuelven a recoger los datos una vez realizada la acción, con estos nuevos datos comprobamos, si realmente se ha realizado la acción o no. Si no se ha podido realizar la acción correctamente, se activa un bit que enciende un led en la RPi suscriptora (dentro del mensaje enviado). Por último, se crea un json con todos los datos relevantes y un id para cada RPi que haga de comedero y se publica indicando en el topic también la id. La ejecución de este programa está configurada para realizarse cada hora con contrab.



Ejemplo de funcionamiento: el detector de humedad nos devuelve un valor de 40 g/m^3 , si nuestro límite es de 50 g/m^3 , el relé se activará activando la bomba. Pero resulta que no hay agua disponible en la reserva. Siguiendo el curso normal del programa, la bomba se apagará pasado un tiempo, entonces el sensor de humedad volverá a devolvernos un valor y será de 40 g/m^3 . Como no ha sido posible rellenar el comedero de agua, ya que el segundo valor recibido sigue siendo inferior a 50 g/m^3 , se enviará un mensaje con el campo de "aviso_agua" a 1. Por último la RPi suscriptora, encenderá la respectiva led de avisos para problemas con el agua, nada más recibir el mensaje.

V. CONCLUSIONES

Mediante la medición de la distancia y la humedad/temperatura hemos conseguido implementar un proyecto de manera que se rellenen el comedero y el bebedero de una ganadería cada cierto tiempo y durante unos segundos, además de avisar al ganadero de cuando los tanques de los suministros no tienen la cantidad suficiente de pienso o de agua.

Este proyecto nos ha ayudado a comprender e integrar mejor los conceptos aprendidos en esta asignatura.

Opinamos que hemos conseguido alcanzar nuestro objetivo, aunque hayamos hecho cambios con respecto a la propuesta inicial. Nos hemos ido adaptando al proyecto, hasta conseguir la mejor implementación posible.

Como grupo, hemos aprendido a rectificar sobre algunos partes del proyecto, como la parte de las alarmas que al principio solo íbamos a hacer una, pero optamos por realizar la alarma para cada tanque individualmente o la elección de las RPi suficientes para realizar el proyecto.

Por otro lado, realizamos una videollamada de forma grupal para concretar que partes hacíamos cada uno, para llevar a cabo la correcta realización del proyecto. Además de comentarnos cualquier problema, duda o cambio por mensaje. Así como, una quedada presencial para grabar el correcto funcionamiento del equipo del ganadero y el equipo de la ganadería.

VI. TRABAJO FUTURO

Si tuviéramos todos los medios posibles para realizar esta solución, se mejoran los siguientes apartados:

Se crearía una aplicación propia para hacer de broker MQTT o se contrataría un servicio estable como [Ubidots](#), para poder realizar la comunicación de manera más fiable, sin caídas espontáneas y con mantenimiento. Ubidots es un broker MQTT que sirve como intermediario a la hora de leer y escribir variables en máquinas y cuenta con un servicio gratuito que ofrece bastantes funcionalidades.

Además, creemos que un sensor de humedad no es viable para la solución. Ya que con días lluviosos daría mediciones por encima de lo habitual y en días áridos por debajo de lo habitual. En algunos de estos casos podría llegar a producir un falso aviso. Sería mejor un medidor de nivel para líquidos.

También se podrían utilizar sensores de mayor calidad que no solo den lecturas de mayor precisión, si no que puedan ser montados a la intemperie y sean más resistentes a golpes. Además, estos sensores deberían ser calibrados en función del comedero en el que se quiera implementar. Cosa que nos lleva a tener que implementar cables de mayor longitud o un sistema IoT más complejo.

El servomotor utilizado, no creemos que sea capaz de abrir y cerrar la tapa del dispensador con suficiente fuerza. Por lo que se tendría que implementar un servomotor más potente, que implica más consumo y probablemente un controlador aparte.

Se utiliza una bomba de agua, accionada mediante un relé. Debería tener en cuenta que la bomba de agua requiere un control bastante importante. Esto es, tendríamos que asegurar el correcto funcionamiento de la misma mediante un sistema que controlará la temperatura de la bomba, durante su funcionamiento y que se asegurará que realmente está bombeando agua.

Por último, la RPI debe estar protegida, ya que trabaja cerca de un sitio con agua. Por lo que habría que diseñar un armario con salidas para los cables que no dejará pasar el agua, entre otras características que debería tener.

Este proyecto empezó siendo una solución bastante económica, pero para que sea realmente viable tenemos que invertir una cantidad de dinero suficiente para hacerla lo más eficiente y robusta posible, ya que va a hacer frente a una condición ambiental adversa.

REFERENCIAS

- [1] G. Eason, B. Noble, and I. N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," *Phil. Trans. Roy. Soc. London*, vol. A247, pp. 529–551, April 1955. (*references*)
- [2] J. Clerk Maxwell, *A Treatise on Electricity and Magnetism*, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
- [3] I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy," in *Magnetism*, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.
- [4] K. Elissa, "Title of paper if known," unpublished.
- [5] R. Nicole, "Title of paper with only first word capitalized," *J. Name Stand. Abbrev.*, in press.
- [6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," *IEEE Transl. J. Magn. Japan*, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].
- [7] M. Young, *The Technical Writer's Handbook*. Mill Valley, CA: University Science, 1989.
- [8] Imagen de tanques, comederos y bebederos
- [9] La Granja Automática. (s.f.). *Dispensador Automático Aníbal*. <https://www.lagranjaautomatica.com/>
- [10] GEA. (s.f.). DairyFeed C 8000 para alimentación automática. *GEA*. <https://www.gea.com/es/products/milking-farming-barn/dairyfeed-feeding-systems/automatic-feeding-dairyfeed.jsp>