

Sensor HX711 conectado al módulo ESP8266 y enviando datos por MQTT a un server JS

Escrito por: Josue Jimenez

www.github.com/Jeziel-Guiev

1. Descripción del proyecto

EL siguiente documento está dedicado a las personas que tienen un conocimiento de electrónica y programación por lo que no se realizara una explicación detallada de cada paso (avisado).

El HX711 es un sensor capaz de medir la fuerza de algún objeto mediante un transductor que para este proyecto es un transductor capaz de medir un peso máximo de 50Kg, pero un sensor puede ser conectado con otros más por lo que si se pone dos transductores el peso máximo de mediación es el doble.

No entraremos en detalles de explicar el funcionamiento de un transductor o el funcionamiento de modulo porque en la red se pude encontrar mucha información, para el proyecto daremos puntos clave del módulo usando su datasheet. El objetivo es realizar la medición del sensor con el módulo ESP8266 y dichos datos serán enviados a un server JS para poder graficarlo y se almacenara si así se lo requiere.



REST API MENV (MongoDB, Express, NodeJS, VueJS)

Materiales

Para los materiales dividiremos en dos secciones, materiales de hardware y software

1. Materiales de hardware

Aquí enlistaremos los materiales necesarios físicos para el proyecto que son:

- Módulo HX711(1 unidad)
- Transductor o célula de carga (2 unidades)
- ESP8266 Wemos D1 mini(1 unidad)
- Necesarios (cables, protoboard, fuente de alimentación si así se requiere)

2. Materiales de software

Para poder visualizar los datos y realizar un análisis de datos o su almacenamiento o todo en conjunto se necesita de un programa para tal tarea, para esta parte se utilizará las siguientes herramientas:

- NodeJs: Servidor Js que será el encargado de realizar una conexión segura con el cliente y con la base de datos, este es el backend.
- VueJs: La parte visual se encargará de conectar al cliente con el servidor por lo tanto Vue Js es nuestro frontend.
- EMQX: El módulo esp8266 necesita enviar los datos a un espacio por este motivo es muy importante establecer porque medio y protocolo se enviará la información, a sabiendas enviar datos de sensores por HTTP no es una manera eficaz ni correcta por lo tanto MQTT es un protocolo ligero y veloz para el mundo IoT EMQX es nuestro server IoT que hace uso de MQTT, será nuestro encargado de enlazar el módulo esp con nuestro servidor y este a la vez con nuestro front end.
- MongoDB: Es nuestra base de datos no relacional quien se encargará de almacenar la información del módulo o módulos e información de usuario.
- Necesarios (Para las gráficas utilizaremos AmchartJS que nos provee un conjunto de gráficos, mqtt, es nuestro cliente mqtt que estará en la parte del cliente, en la parte del servidor no se dará muchos detalles de su desarrollo si bien puede clonarlo directamente desde el repositorio de GitHub. Todo el programa necesita estar alojado en algún lugar este puede ser una raspberry Pi, una PC, o un server cloud para este proyecto lo ejecutaremos en un server cloud de la empresa DigitalOcean, pero para las primeras pruebas lo ejecutaremos en la PC).

2. Desarrollo

¡Manos a la obra!, empezaremos por el circuito en su conexión con el módulo y los transceptores posteriormente seguiremos con su programación con el entorno de Arduino IDE para poder ver en puerto serial los datos de la medición, seguido lo configuraremos para poder conectarse a una red para poder proveerle de internet y estableceremos una conexión con EMQX de manera independiente sin tocar nada del backend y frontend por último realizaremos los últimos dos pasos mencionados.

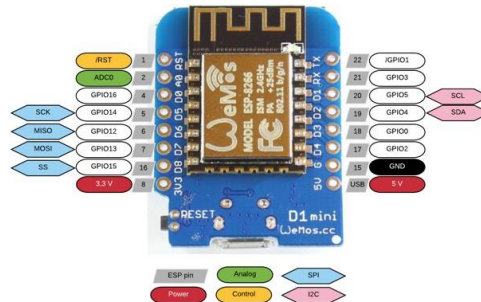
Diagrama del circuito

- El transductor que utilizaremos es:



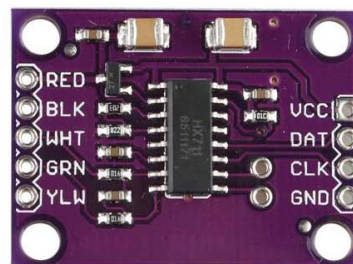
Cable rojo 5v, cable negro GND, cable blanco OUTPUT de información

- ESP8266 wemos



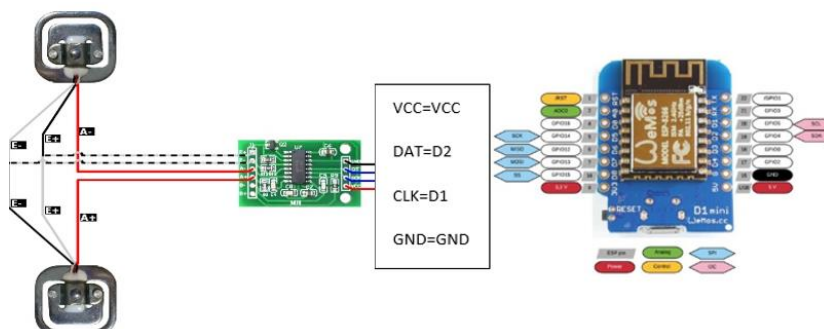
Microcontrolador con antena Wifi

- HX711



Los pines de la derecha estarán conectadas al módulo ESP8266, mientras que los pines de la izquierda irán a los transductores

- Conexión



Los pines se deben conectar con su igual que están en la placa es, vcc con su vcc

El valor que le lanza es 0.3Kg, pero nuestro material es de 1Kg por lo que el factor de calibración nos ayudara a resolver. En mi caso estoy utilizando Putty para la comunicación serial. Al apretar a o z nuestro factor de calibración cambiara al asigna "a" nuestro factor bajara aumentando así el peso medido y z funcionaria al revés, en mi caso tuve que bajarlo hasta el siguiente factor de calibración.

Referencias bibliográficas

- ❖ <https://learn.sparkfun.com/tutorials/load-cell-amplifier-hx711-breakout-hookup-guide/all#combinator>.
- ❖ https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711_english.pdf