



Universidad
Zaragoza

Trabajo Fin de Grado

Sistema de monitorización IoT de datos de una
bomba de agua para su mantenimiento predictivo

IoT data monitoring system of a water pump for
predictive maintenance

Autor

Pablo Gallego Gimeno

Director/es

Roberto José Casas Nebra

Alberto Mur López

Escuela de Ingeniería y Arquitectura

Universidad de Zaragoza

Curso 2019/2020

RESUMEN

El presente trabajo consiste en el desarrollo de un sistema IoT para el análisis de la eficiencia y el mantenimiento predictivo de una instalación de captación de aguas subterráneas. Para llevar a cabo dicho estudio se debe realizar la sensorización de la instalación con la finalidad de monitorizar a través de internet las variables e indicadores clave de eficiencia.

Se empezará por analizar la problemática detrás de la eficiencia energética en estas instalaciones y determinar cuáles son los parámetros que se deben medir y controlar. Después se pasará a elegir los sensores que permitan recoger la información necesaria. Con los datos recogidos de los sensores, se elaborará una algoritmia a partir de estos que permita obtener una serie de indicadores clave de eficiencia que se han definido.

Para implementar el programa se va a utilizar una pasarela inteligente Siemens Simatic IoT2040, adecuada para trabajar en un entorno industrial y en este tipo de aplicaciones del internet de las cosas. Esta plataforma estará conectada a internet y se programará en Node-RED, una herramienta de desarrollo basada en flujos para una programación visual, orientada a aplicaciones IoT. Con esto se podrá obtener la información y desarrollar la algoritmia necesaria para calcular los indicadores clave de la eficiencia de la instalación. Los datos de nuestro análisis se recogen mediante el protocolo MQTT de una instalación de bombeo real.

Para el análisis y mantenimiento predictivo, se pasará a enviar los resultados de la algoritmia aplicada a una interfaz gráfica local tipo SCADA donde se visualizarán todos los datos en diferentes pantallas, y, a su vez, se enviarán mediante el protocolo MQTT a un servidor para su almacenamiento a largo plazo en una instancia InfluxDB, que es una base de datos de series temporales. Todo esto se realiza continuamente en tiempo real, por lo que será más eficiente que una medición puntual a la hora de realizar un análisis de la instalación. En caso de que un parámetro o KPI del sistema esté fuera de rango, se notificará automáticamente mediante un correo electrónico.

ÍNDICE

1. Introducción	1
1.1 Marco del proyecto	1
1.2 Objetivos del trabajo	2
1.3 Estudio de la problemática	4
2. Sensorización	8
2.1 Tipos de sensores para cada parámetro	8
2.1.1 Nivel de agua	8
2.1.2 Caudal	8
2.1.3 Presión	9
2.1.4 Datos eléctricos	9
2.1.5 Temperatura del motor	10
2.1.6 Horas de funcionamiento	10
2.2 Tipos de entrada de sensores	11
3. Diseño de la algoritmia	12
3.1 Definición de los KPIs	12
3.2 Obtención de los KPIs	12
3.3 Interpretación de los KPIs	15
3.4 Aplicación a casos reales	16
4. Plataforma IoT	17
4.1 Análisis de alternativas	17
4.2 Elección de la plataforma	21
4.3 Preparación del entorno	22
4.3.1 Node-RED	22
4.4 Flujos de lectura de datos de sensores	22
4.4.1 Esquemático en <i>Circuit Maker</i>	27

4.5 Flujos de recogida de datos de la web	28
4.6 Flujos de cálculo de indicadores	29
4.7 Notificación por correo electrónico de parámetros o KPIs fuera de rango	30
5. Pruebas en entorno real	32
5.1 Descripción del escenario de pruebas	32
5.2 Sensorización y recogida de datos	33
5.3 Monitorización y envío a base de datos	36
5.3.1 Flujos de interfaz gráfica local	36
6. Conclusiones y trabajo futuro	41
6.1 Posible instalación del sistema	41
6.2 Conclusiones personales	42
7. Bibliografía	43
8. Anexos	45
8.1 ANEXO 1: Elección de sensores y justificación	45
8.2 ANEXO 2: Aplicación de algoritmia a casos reales	47
8.3 ANEXO 3: Configuración de la conexión del IoT2040	53
8.4 ANEXO 4: Instalación de paquetes en Node-RED	56
8.5 ANEXO 5: Conexión y programación de los sensores	58
8.6 ANEXO 6: Código en JavaScript de los nodos en Node-RED	67

LISTA DE FIGURAS

Figura 1 Evolución de la industria	1
Figura 2 Representación del IoT	2
Figura 3 Arquitectura del sistema	3
Figura 4 Cronograma del trabajo	3
Figura 5 Instalación de sondeo	4
Figura 6 Representación de descensos	5
Figura 7 Bomba de agua	5
Figura 8 Curvas nominales de la bomba	6
Figura 9 Representación gráfica de Selección de bomba	15
Figura 10 Raspberry Pi 3 B+	17
Figura 11 M-Duino	17
Figura 12 Rijing Fogs7	18
Figura 13 Flexy 205	18
Figura 14 Conprosys Nano	18
Figura 15 Babelgate G5001	18
Figura 16 Simatic IoT2040	19
Figura 17 FP-I4C	19
Figura 18 Avtse-RPI-IIOTG	19
Figura 19 Tscontrol	20
Figura 20 Nio-101	20
Figura 21 Modulo I/O en Simatic IoT2040	21
Figura 22 Logo de Node-RED	22
Figura 23 Sensor de pulsos	23
Figura 24 Flujo de Node-RED para medir el caudal	24
Figura 25 Piezómetro	24
Figura 26 Flujo de Node-RED para la medición del nivel de agua	24
Figura 27 Analizador de redes	25
Figura 28 Flujo de Node-RED para la medición de datos eléctricos	25
Figura 29 Sensor de presión G1/4	26
Figura 30 Flujo de Node-RED para la lectura del sensor de presión	26
Figura 31 Termistor NTC	26
Figura 32 Flujo de Node-RED para la lectura del termistor	26

Figura 33 Flujo de Node-RED para los sensores	27
Figura 34 Esquemático del circuito	28
Figura 35 Flujo de Node-RED para recoger un dato desde web	28
Figura 36 Flujo de Node-RED para recogida de todos los parámetros desde la web	29
Figura 37 Flujo de Node-RED para el cálculo de los KPIs	30
Figura 38 Flujo de envío de datos fuera de rango vía e-mail	31
Figura 39 Bandeja de entrada del correo electrónico con mensajes de aviso	31
Figura 40 Ensayo de bombeo escalonado en la instalación real	32
Figura 41 Curva Q-H de la bomba	32
Figura 42 Mediciones en pozo real	33
Figura 43 Datos teóricos y KPIs de mediciones reales	33
Figura 44 Arquitectura del protocolo MQTT	34
Figura 45 Datos de la instalación real	34
Figura 46 Flujo para recoger datos de un fichero CSV	35
Figura 47 Flujo de recogida de datos de instalación real vía MQTT	35
Figura 48 Flujo de Node-RED para representar los parámetros en la dashboard	36
Figura 49 Flujo de Node-RED para representar los KPIs en la dashboard	36
Figura 50 Interfaz gráfica de parámetros en tiempo real	37
Figura 51 Interfaz gráfica de KPIs en tiempo real	38
Figura 52 Interfaz gráfica de la evolución de los parámetros en el tiempo	38
Figura 53 Nodo de envío MQTT de Node-RED	39
Figura 54 Monitorización en base de datos influxDB	39
Figura 55 Interfaz de la base de datos y monitorización	40

1. Introducción

1.1 Marco del proyecto

La eficiencia y el mantenimiento predictivo de las instalaciones de captación de agua son factores muy importantes atendiendo al ahorro energético y económico que puede haber detrás. Nos ayudaremos del uso de nuevas tecnologías para diseñar el sistema, teniendo de base el concepto de “el internet de las cosas”.

Este proyecto está basado en el **IoT** (uno de los conceptos clave de la **industria 4.0**) ya que queremos monitorizar y analizar los datos recogidos de una pasarela inteligente mediante una plataforma online que envíe información a través de internet. Para entender mejor este concepto, vamos a empezar explicando qué es la industria 4.0 y el internet de las cosas:

La industria 4.0 consiste en la transformación digital de la industria, lo que se considera ya como la **cuarta revolución industrial**. Esto supone la inclusión de nuevas tecnologías en la industria para hacerla más “inteligente”, con cambios importantes en las condiciones tecnológicas de producción.



Figura 1 Evolución de la industria

Estas nuevas tecnologías de software inteligente, sensores y procesamiento de datos tienen como finalidad automatizar y hacer más efectivos los procesos productivos. Una de las bases de esta nueva industria es la conexión a internet de todas las cosas, el denominado IoT, que se explicará a continuación.

El internet de las cosas (también llamado IoT por sus siglas en inglés: Internet of Things) es un concepto relativamente nuevo que consiste en la conexión de objetos a la red con el objetivo de que todo acabe estando conectado para poder controlarse remotamente. Este concepto puede abordar desde el control de sensores, actuadores u otro tipo de dispositivos electrónicos, hasta la conexión de objetos cotidianos a la red; todo con la finalidad de hacer más cómoda la vida de las personas.

El IoT no se trata del futuro, sino más bien del presente, ya que en la actualidad más dispositivos apuestan por esta tecnología que está cada vez más presente tanto en el ámbito industrial como en la vida cotidiana; esto se debe a que sus aplicaciones son prácticamente ilimitadas. Esta tecnología está en continua evolución y parece que va a ser imparable. Es y será un cambio

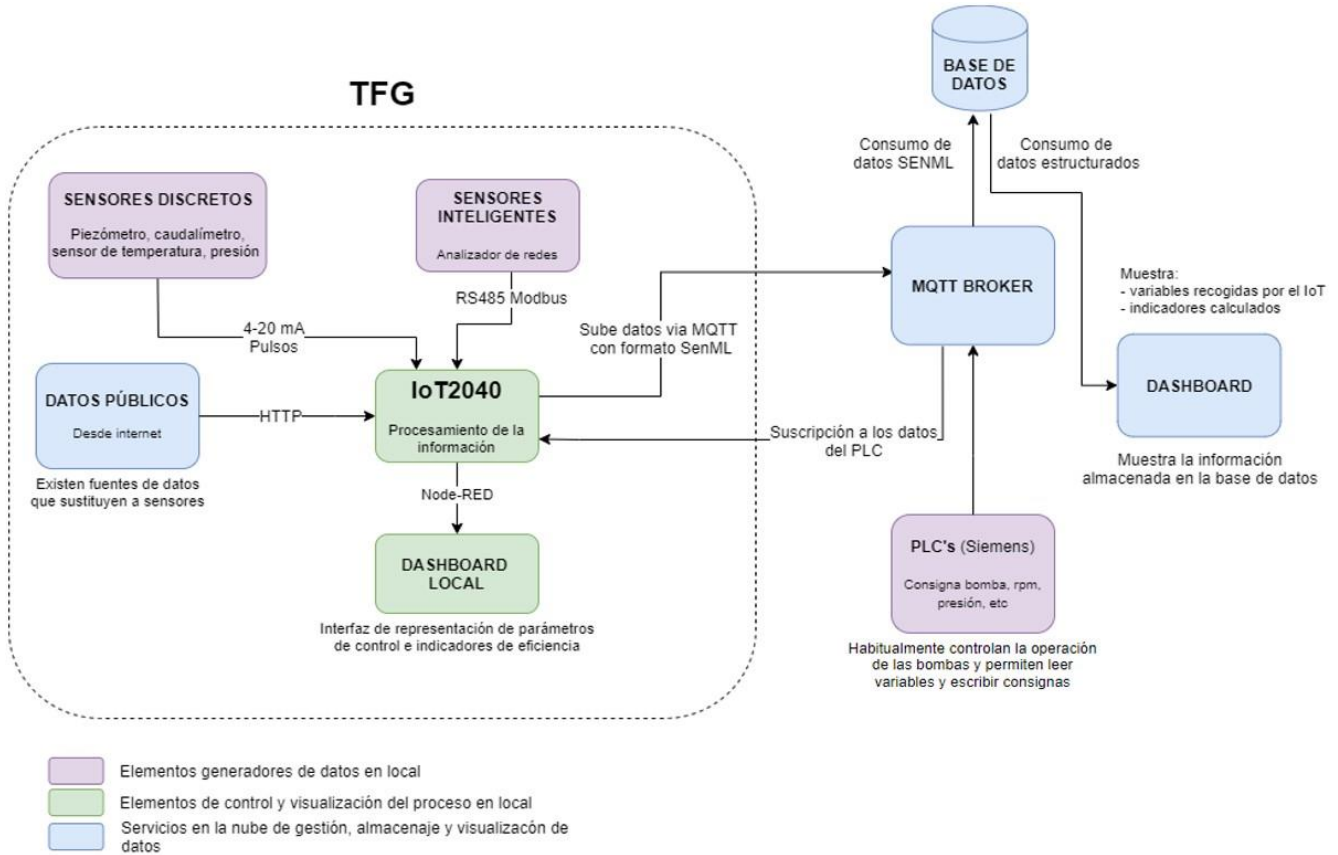


Figura 3 Arquitectura del sistema

A continuación se representa el cronograma donde aparece el tiempo dedicado a cada fase del trabajo:

TAREAS	Noviembre	Diciembre	Enero	Febrero	Marzo	Abril	Mayo	Junio
Reuniones iniciales	[Barra]							
Investigación sobre la problemática		[Barra]						
Búsqueda de sensores			[Barra]					
Diseño de la algoritmia				[Barra]				
Alternativas de plataforma				[Barra]				
Programación de algoritmia					[Barra]			
Recogida de datos de sensores					[Barra]			
Recogida de datos de web						[Barra]		
Envío a interfaz gráfica							[Barra]	
Envío a base de datos MQTT								[Barra]
Notificaciones vía e-mail								[Barra]
Recogida de datos reales MQTT								[Barra]
Redacción de la memoria		[Barra]						

Figura 4 Cronograma del trabajo

1.3 Estudio de la problemática

La eficiencia energética en este tipo de instalaciones de bombeo de agua es un factor muy importante. Para empezar, vamos a explicar brevemente cómo se captan las aguas subterráneas mediante el bombeo del agua:

En los acuíferos de agua natural, que representan alrededor del 30% del agua dulce de la tierra, el agua procedente de las precipitaciones se filtra por debajo del suelo hasta generar estos pozos subterráneos. Para poder sacar esta agua se requiere una perforación para la instalación de bombas. Es un sistema compuesto por tres elementos relacionados entre sí, sus propiedades deben considerarse de manera conjunta e integrada y no sólo como la simple suma de sus partes: acuífero, sondeo y equipamiento electromecánico.^[1]

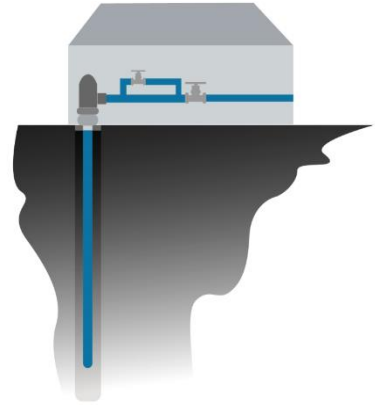


Figura 5 Instalación de sondeo

- **Acuífero:** en geología, un acuífero es un conjunto de rocas que permiten la permeabilidad del agua y la pueden acumular en sus poros o grietas, por lo tanto, son reservas de agua subterráneas.
- **Sondeo:** se refiere a las perforaciones realizadas desde la superficie para llegar a las aguas subterráneas. Son estrechos y pueden llegar a tener hasta cientos de metros de profundidad.
- **Equipamiento electromecánico:** todos los elementos necesarios para bombear el agua hacia la superficie, debido a esto, el componente principal será la bomba.

Son sistemas que tienen un consumo energético muy elevado, por lo que se tiene que conseguir un buen rendimiento para así mejorar el funcionamiento y no encarecer el proceso.

Según los expertos de Grundfos (fabricante de bombas hidráulicas, referente en bombeo, innovación y tecnología del agua), la captación de éste agua supone hasta el 50% de la electricidad invertida en el suministro de agua. Los costes eléctricos suponen un 85% del total, por lo que es ahí donde interesa optimizar el proceso. Un consumo eléctrico excesivo provoca que el precio por litro de agua extraído haga replantear la rentabilidad de la instalación.

El principal requerimiento del caudal de explotación va a ser que sea el suficiente para abastecer la demanda de agua diaria.

El principal problema es que el sistema no trabaja de forma ideal, sino que existirán unas pérdidas. Por lo tanto, la eficiencia energética total será la suma de la eficiencia de la instalación electromecánica más la eficiencia referida a la diferencia entre el descenso teórico y el descenso real del acuífero, eficiencia hidráulica.

Existirá por lo tanto una necesidad de obtener unos indicadores y de un seguimiento, control y diagnóstico de todo esto para su optimización energética. Primero vamos a ver la reducción de la eficiencia debida a la diferencia entre el descenso teórico y el descenso real en los acuíferos.

El descenso teórico será una función directamente proporcional al caudal, mientras que el descenso real corresponde al observado en el sondeo. La diferencia entre estos dos va a ser las pérdidas que representan varias causas:

- Penetración a través del relleno de grava y de las rejillas del entubado.
- La fricción del agua con las paredes del tubo.
- Pérdidas por la no validez de la ley de Darcy (ley que escribe, con base en experimentos de laboratorio, las características del movimiento del agua a través de un medio poroso, válida para medios con un número de Reynolds $Re < 1$) debido al posible establecimiento de un régimen turbulento por excesiva velocidad del agua en las proximidades de la zona donde se realiza el sondeo.
- Pérdidas en la bomba y el mismo sondeo si la instalación está mal diseñada o en malas condiciones.

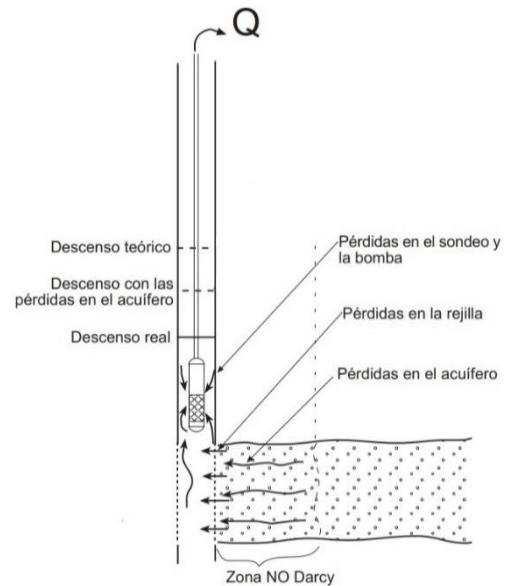


Figura 6 Representación de descensos

Por lo tanto, el descenso real siempre va a ser superior al teórico.^[2]

Por otra parte, hay que estudiar la eficiencia en la instalación electromecánica. Una bomba de agua centrífuga tiene como función trasladar agua de un sitio a otro, su funcionamiento es el siguiente:

Primero el agua entra en la bomba por su entrada y mediante un motor, impulsa el agua que se evacúa por su salida. Los motores de estas bombas transforman la energía eléctrica en mecánica y funcionan como el resto de los motores eléctricos, mediante la acción de los campos magnéticos generados en sus bobinas, compuestos por un rotor y un estator.

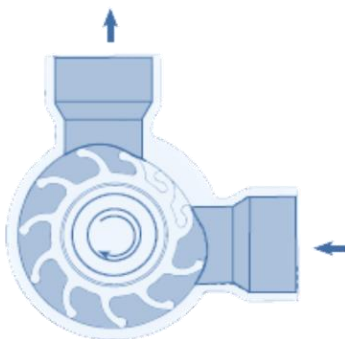


Figura 7 Bomba de agua

Estos motores tienen sus pérdidas, que pueden ser tanto dependientes de la carga como independientes. Si nos fijamos en las dependientes nos encontramos con las pérdidas de cobre en el inducido, pérdidas mecánicas por rozamiento y fricción y otras pérdidas misceláneas. En las independientes a la carga están las pérdidas por histéresis y las pérdidas por corrientes de Foucault.^[3]

En definitiva, la eficiencia de la instalación electromecánica será la relación entre el caudal bombeado y el consumo energético.

La calidad química del agua también interviene en la eficiencia del sistema. El grado de incrustabilidad (en la mayoría de las ocasiones ligado a la concentración de Ca y Mg) incide en la durabilidad de los equipos de bombeo, afectando también a la eficiencia del pozo. De la misma forma, un potencial redox negativo en el agua del acuífero se convierte en positivo al poner el mismo en contacto con la atmósfera, produciendo oxidaciones. Además, si el acuífero es multicapa (son un caso particular de acuíferos en los que se suceden niveles de distinta permeabilidad) ^[4] y con aguas de diferente conductividad, se crean corrientes eléctricas en la conducción y sobre todo a ambos lados de la aspiración de la bomba, que inciden también en su vida útil de forma importante.

Todas estas pérdidas afectan al rendimiento del sistema, y cuanto mayores sean, menor va a ser la eficiencia energética total.

Además de todas las pérdidas vistas anteriormente, los equipos van a tener una vida útil limitada, debido al desgaste o la antigüedad. Esta duración va a depender entre otras cosas de la corrosión del agua, el entorno en el que trabajan, el promedio de arranques diarios, desgaste, etc. Con la avería de una bomba, hay que hacer balance de si es rentable repararla o adquirir una nueva, además de la parada de la instalación, con las pérdidas que esto supone.

Para ver el desgaste y si fuese oportuno la elección de la bomba adecuada para la instalación, se tiene que definir el caudal (Q) a bombear y la altura manométrica (H) a la que va a trabajar el equipo. La altura manométrica expresada en metros de columna de agua (m.c.a.) corresponde a la altura de aspiración de columna de agua con la que debe trabajar la bomba.^[5]

Habrá que analizar si se debe cambiar la bomba o no en función de lo alejado que esté el punto de operación Q-H del punto BEP (Best Efficiency Point, punto de máxima eficiencia), donde es más óptimo el sistema. Para esta bomba se localiza el caudal al cual se obtiene el máximo rendimiento y se ajusta el nivel de agua necesario para tener ese valor de caudal. Gráficamente: curvas nominales de la bomba, donde la línea azul se corresponde con la curva característica Q-H, o curva de nivel de agua de la instalación, siendo Q el caudal y H la altura manométrica de agua; representa la altura total requerida por la instalación en función del caudal. La curva de color rojo es el rendimiento del sistema, teniendo en cuenta las pérdidas, en función del caudal.

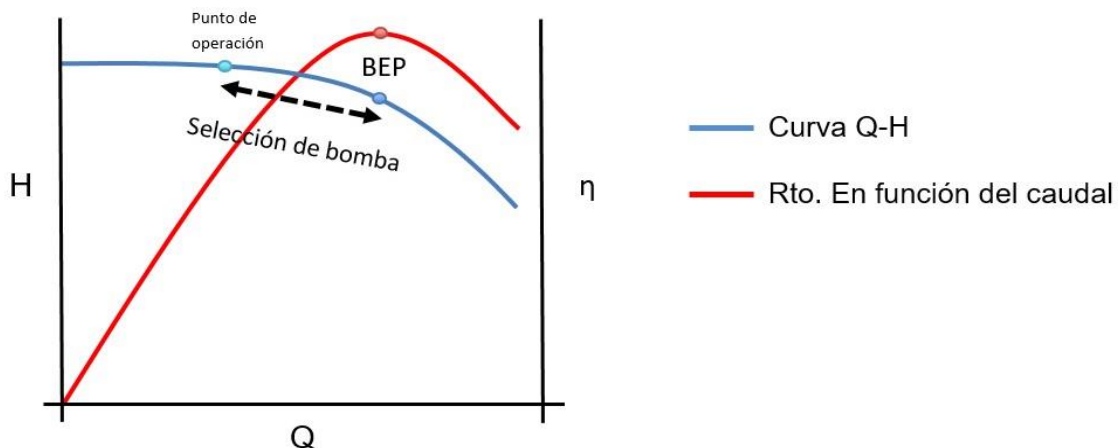


Figura 8 Curvas nominales de la bomba

Estas curvas dependen también de la potencia aplicada de la bomba, su velocidad. Por lo tanto, cambiando la velocidad mediante un variador de frecuencia, el punto de operación también será diferente.

Como se ha comentado anteriormente, hay que buscar optimizar el proceso y lograr una eficiencia lo más elevada posible, además de poder prevenir posibles ineficiencias. Para conseguirlo, se tendrá que realizar un estudio de la eficiencia del sistema de captación de aguas subterráneas. Va a ser muy importante el análisis de los resultados que tomemos del proceso y su control, ya que las acciones que se tomarán pueden acabar suponiendo un ahorro de miles de euros mensuales en alguna instalación.

Para todo este análisis y estudio de la eficiencia, el primer paso deberá ser conocer las variables necesarias de las cuales dependa la eficiencia de nuestro sistema. Empezaremos por definir los datos que habrá que tener en cuenta, por un lado, los que se van a medir: ^[1]

- **Profundidad del nivel de agua** (m): nivel de agua del acuífero que da información de su estado, midiendo los niveles estático y dinámico se obtiene el descenso.
- **Caudal** (l/s): volumen de agua por unidad de tiempo, afecta al rendimiento.
- **Presión** (m.c.a.): factor que informará de la altura de agua a aspirar.
- **Datos eléctricos** sin carga y con carga:
 - Tensión (V)
 - Corriente (A)
 - Potencia activa, P (kW)
 - Potencia reactiva, Q (kVAr)
 - Potencia aparente, S (kVA)
 - Factor de potencia, $\cos\phi$

El más relevante en cuanto al análisis del rendimiento de la bomba será la potencia activa consumida (kW).

- **Temperatura del motor** (°C): el calentamiento de la bomba debe controlarse.
- **Tiempo de funcionamiento** (h): tiempo que ha estado activo el sistema.
- **Parámetros químicos**: es importante realizar análisis químicos periódicos de turbidez, conductividad, arrastres, etc.
- **Otros** (Precipitación, volumen de embalses, nivel de piezómetros cercanos, etc.): más parámetros que afectan directamente al estado de las aguas subterráneas, por lo tanto, al sistema.

Por otro lado, se tendrán que conocer los valores correspondientes a las características de la instalación: ecuación del pozo, caudales teóricos, valores nominales de potencia y rendimiento de la bomba.

Posteriormente, se expondrá la importancia de cada parámetro y cómo obtenerlos. Como conclusión, para mejorar la eficiencia de la explotación de aguas subterráneas, va a ser indispensable tener el control de unos determinados parámetros para ajustar el sistema y optimizarlo, así como cumplir las necesidades de diagnóstico, control y seguimiento adecuadas. Esto pasa por la inclusión de sensores en el sistema para poder obtener toda esa información.

2. Sensorización

Como hemos explicado, tendremos que recurrir al uso de sensores para poder analizar la eficiencia del sistema. Vamos a pasar a definir qué tipo de sensores necesitamos para este caso.

2.1 Tipos de sensores para cada parámetro

Empezaremos analizando la importancia y la necesidad de medir cada uno de los parámetros explicados anteriormente y los diferentes sensores para cada parámetro.

2.1.1 Nivel de agua

El nivel de agua del acuífero (medida en metros) es un parámetro esencial que debemos medir para conocer la cantidad de agua que hay y el estado del acuífero, y por tanto determina en parte el equipo que usar.

Para medir el nivel de agua en la instalación se pueden usar dispositivos como un tubo piezométrico o una sonda manual.

- El **tubo piezométrico** consiste en un tubo sale de la zona con agua (por un lateral) hacia arriba y por el cual el agua sube hasta una altura equivalente a la presión de ésta en el punto donde empieza el tubo. A partir de la presión obtenida y la altura del tubo se puede conocer el nivel de agua.^[6]
- Las **sondas de nivel** son las más comunes para medir en este tipo de casos. La sonda da señal cuando ésta entra en contacto con el líquido, así se conoce la profundidad. Son sistemas muy robustos y sencillos que además pueden llegar a trabajar a grandes distancias y profundidades.
- **iDrolewell** es un sistema de medida de nivel de agua en sondeos, basado en el principio de medida por burbujeo. Es muy robusto y preciso y de larga durabilidad, siendo sondas de medición en continuo.^[7]
- También existen sondas que miden a través del uso de ultrasonidos, ya que esta tecnología usada para detección o medir distancias también es aplicable en el caso de medir el nivel de agua. No se suelen utilizar para este tipo de aplicaciones debido a que las perforaciones suelen tener unos grados de desviación de la vertical, en sondeos profundos esta tecnología no valdría.

2.1.2 Caudal

El caudal del agua producida es también otro parámetro que será necesario analizar ya que de éste depende directamente el rendimiento de la bomba. Realmente es el dato más importante de la instalación, representando la cantidad de agua extraída por unidad de tiempo. Para medirlo se pueden utilizar distintos dispositivos (caudalímetros o contadores):

- **Caudalímetro electromagnético**: su funcionamiento se basa en la Ley de inducción de Faraday, donde la tensión es proporcional al producto de la velocidad del conductor por

la intensidad del campo electromagnético y por la longitud del conductor; En este caso, como el agua es conductora, la velocidad será la del agua y la longitud es la distancia entre unos electrodos colocados en el agua. Al establecer un campo magnético perpendicular al sentido de la circulación, tenemos una tensión proporcional a la velocidad del agua.^[8]

- **Contadores volumétricos:** atrapan una cantidad de agua fijada y se traslada a la salida. Poco habituales pero muy precisos.
- **Contadores de la velocidad del agua:** a partir de la estimación de la velocidad del agua se obtiene el valor del caudal. Son los más baratos y tienen una exactitud aceptable.
- **Contador Woltman o de turbina:** funciona con una turbina dentro de la tubería cuyo eje es paralelo al flujo de agua; Ésta mediante imanes acciona los engranajes que accionan el contador.^[9]
- **Caudalímetro por ultrasonidos:** son sensores externos que envían ondas ultrasónicas a través del agua para determinar su velocidad. También miden la temperatura del agua para ajustarse al cambio de densidad de esta con la temperatura. Son bastante precisos pero caros.

Se requiere que el sensor de caudal elegido sea capaz de medir rangos de 0 a 100 l/s.

2.1.3 Presión

La presión hidrostática es una forma indirecta de tener información sobre la altura manométrica de aspiración, midiendo una vez que el agua llega a la superficie. Además, es un factor meteorológico que puede tener efecto sobre las aguas subterráneas. Para medirlo se pueden utilizar:

- **Transmisores o sensores de presión:** son sensores la lectura de un detector de presión a una señal de salida.
- **Presostatos:** son dispositivos que se activan al alcanzarse un determinado valor de presión.
- **Manómetros digitales:** sacan la medida de un sensor de presión en una pantalla.
- **Manómetros de tubo de Bourdon:** cuando la presión actúa sobre un material elástico, produce un movimiento que será la medida.

Se requiere un sensor que mida presiones de agua distintas a la atmosférica, pero del mismo rango, además debe ser resistente para trabajar en ambientes como el de nuestro caso.

2.1.4 Datos eléctricos

Respecto a los datos eléctricos que se deben conocer para realizar el control, son parámetros fundamentales que se analizan para evitar daños en la instalación eléctrica y en los equipos. Para recogerlos lo adecuado será recurrir al uso de un analizador de redes que pueda mostrar la tensión, corriente, potencia y factor de potencia.

El analizador de redes es un instrumento que permite analizar una gran variedad de parámetros eléctricos para poder llevar un control y gestión de un sistema.

2.1.5 Temperatura del motor

La temperatura del motor es otro parámetro que se tiene que controlar ya que afecta directamente a la eficiencia de este. Un aumento de la temperatura por encima de la nominal supone un incremento lineal de pérdidas de carga en el motor. Por lo tanto, si queremos comprobar que no existen sobrecalentamientos y que funciona de forma correcta, será necesario conocerla.

- El **sensor o sonda de temperatura** detecta variaciones de ésta y la convierte en una señal eléctrica. Para medir la temperatura existen varios tipos de sensores:^[10]
- **Termopar**: su funcionamiento consiste en ver la variación de tensión de dos cables de metales distintos unidos en un extremo. Son los más utilizados en el entorno industrial, robustos, pequeños y baratos.
- **Termistor**: son materiales semiconductores cuya resistencia varía con la temperatura, usan electrodos internos que detectan y miden el calor a través de impulsos eléctricos.
 - NTC** (Negative Temperature Coefficient): a más temperatura, menor resistencia.
 - PTC** (Positive Temperature Coefficient): a mayor temperatura, mayor resistencia.
- **Infrarrojo**: es un tipo de sensor sin contacto que mide la radiación infrarroja que emite un cuerpo. Se utilizan para medir objetos inaccesibles o en movimiento. Son baratos y de buena calidad.
- **RTD** (Resistive Temperature Device): su funcionamiento es similar al del termistor, modificando su resistencia con la temperatura, pero preparados para medir temperaturas más elevadas. Se compone de un alambre enrollado con un núcleo de vidrio o cerámica que lo rodea. Son muy precisos pero caros.

2.1.6 Horas de funcionamiento

También se tendrá que llevar el control de las horas de funcionamiento de la instalación, para lo cual se puede utilizar un contador de horas. Se realizará por software, llevando el control del tiempo que lleva activo el sistema mediante el mismo programa.

Los sensores recogerán las medidas que posteriormente serán tratadas y comparadas con los resultados teóricos para poder tomar acciones y así mejorar la eficiencia energética y optimizar el sistema. Con esto se pueden recopilar datos históricos, ver la evolución de los parámetros con el tiempo y así poder llegar a conclusiones como puede ser el fallo de un equipo.

2.2 Tipos de entrada de sensores

Ahora explicaremos las posibles interfaces que nos encontramos según la entrada de los datos de los sensores al controlador:

- **Sensores con salida analógica:** proporcionan una señal analógica que puede ser de tensión o corriente. Los valores más utilizados en estos casos son señales de 0 a 10 V o de 4 a 20 mA.
- **Sensores con señal de salida digital:** la salida es digital, con valores de todo o nada, '1' o '0', y las tensiones más comunes son 5 o 24V.
- **Comunicación serial:** los sensores con estas salidas transmiten información de forma serial (trama de bits) enviando datos al controlador por medio de un puerto de comunicaciones. Son para comunicarse por buses de datos como pueden ser RS232 o RS485.
- **Procedentes de la nube:** otra opción para que la información de los parámetros llegue al controlador puede ser a través de la nube, recogiendo datos en tiempo real de una plataforma online o de una base de datos.
- **Procedentes de un PLC:** muchas veces la opción es recoger la información del PLC que controla el sistema, pudiendo leer variables y escribir consignas.

En el **ANEXO 1** se exponen los sensores elegidos para medir cada parámetro y su justificación.

3. Diseño de la algoritmia

3.1 Definición de los KPIs

A partir de los parámetros obtenidos con los sensores se pueden construir unos indicadores mediante los cuales entender mejor el comportamiento del sistema.

Para realizar el análisis deseado tenemos que fijarnos en unos indicadores clave determinados. Estos parámetros que se deben controlar no se obtienen directamente de la lectura de los sensores, sino que habrá que diseñar los algoritmos necesarios para obtenerlos. Los indicadores clave de rendimiento o KPI (Key Performance Indicator) de la instalación son: ^[1]

Eficiencia hidráulica pozo-acuífero:

- **Caudal actual vs caudal recomendable:** informa sobre el incremento de descensos en régimen turbulento.
- **Caudal específico (l/s/m):** da información sobre colmatación, incrustación en filtros y rejillas. Es un indicador muy interesante, ya que puede dar una lectura indirecta de las pérdidas de carga. El nivel dinámico (por lo tanto, el descenso) es muy variable, pero el caudal específico será más estable y las tendencias en su valor son muy buenos indicadores de pérdidas de eficiencia en la captación.
- **Eficiencia hidráulica del pozo (%):** nos informa sobre cambios de posición del nivel piezométrico y nivel regional en el acuífero. Diferencia entre el descenso teórico y el real.

Eficiencia en equipamiento de impulsión:

- **Rendimiento real de la bomba (%):** representa la eficiencia de la bomba respecto al consumo de potencia teórico y el real.
- **Desviación rendimiento total (%), rto. real vs teórico:** indica el desgaste en la bomba.
- **Selección bomba (%):** informa si el punto Q-H está alejado del BEP o punto de máxima eficiencia.
- **Ratio consumo unitario kWh/m³:** indica si hay sobreconsumo por un bajo rendimiento total, si hay problemas con el sistema electromecánico.
- **Ratio de eficiencia Wh/m³/m:** puede ser adecuado o no. Indica si la electrobomba trabaja de forma ineficiente.

3.2 Obtención de los KPIs

La obtención de los parámetros de los sensores es el medio para calcular los indicadores anteriores experimentalmente. Se tiene que desarrollar una algoritmia cuya entrada sea el valor de las lecturas de los sensores y sea capaz de calcular el valor de cada KPI del sistema.

Para la obtención del resultado de la **comparación entre el caudal actual y el caudal recomendable** necesitamos la medida del caudalímetro y el valor establecido del caudal

recomendable, para el cual tenemos un mayor rendimiento de la bomba. Se tiene que obtener la diferencia entre ambos mediante su resta. Este resultado tendrá que ser lo más pequeño posible.

$$Diferencia = Q_{medido} - Q_{recomendable}$$

En este caso el **caudal específico** es el caudal por unidad de longitud (l/s/m). La longitud que utilizamos se corresponde al descenso, d (en metros). Por lo tanto, el caudal específico lo sacaremos a partir de la siguiente fórmula:

$$Q_e = \frac{Q_{medido}}{d}$$

Un indicador clave va a ser la **eficiencia hidráulica** del pozo. Para la obtención de ésta tenemos que calcular el cociente entre el descenso teórico y el descenso real. El descenso real se corresponde a la diferencia entre la medición del nivel dinámico y el estático, que son:

- Nivel estático: se refiere al nivel del pozo cuando la bomba está detenida y partiendo del reposo. Se ve afectado por el nivel regional del acuífero principalmente, cuya estacionalidad o tendencias son variables en cada caso; también por efectos meteorológicos y demás factores externos en menor medida.
- Nivel dinámico: es el nivel de agua cuando la bomba está en funcionamiento.

$$d_{real} = Nivel\ dinámico - Nivel\ estático$$

El descenso teórico es una función lineal del caudal que vamos a representar como: ^[2]

$$d_{teórico} = A * Q_{medido}$$

El descenso real será la suma de las pérdidas producidas por varias causas explicadas previamente al descenso teórico:

$$d_{real} = A * Q_{medido} + B * Q_{medido}^n$$

Para obtener los valores de A , B y n , se tendrá que medir el descenso real a unos distintos caudales establecidos y despejar. Los coeficientes se determinan experimentalmente a partir de ensayos de bombeo escalonados. Es conveniente disponer de al menos tres escalones.

El caudal en la fórmula estará expresado en $m^3/día$. Con esto sacamos el descenso teórico para pasar a obtener la eficiencia.

La eficiencia hidráulica vendrá dada por la siguiente fórmula:

$$Eficiencia\ hidráulica = \frac{d_{teórico}}{d_{real}} * 100$$

El **rendimiento real de la bomba** se va a representar como el cociente entre la potencia teórica característica de la bomba y la consumida real. Para poder satisfacer las necesidades de caudal y altura, la potencia real tendrá que ser mayor que la teórica. La potencia teórica se saca a partir del caudal (Q, en l/s) y la altura manométrica (H, en m.c.a.):

$$P_{teórica} = \frac{0.736xQxH}{75}$$

Y con esto se obtiene el rendimiento:

$$\eta_{real} = \frac{P_{teórica}}{P_{real}} * 100$$

La **desviación del rendimiento real frente al teórico** de la bomba es la diferencia entre ambos porcentajes expresada en %. El rendimiento teórico es un parámetro conocido de la bomba que depende del caudal bombeado, mientras que el rendimiento real es el cociente entre la potencia teórica (nominal) de la bomba de agua y la potencia real (medida). Para obtenerla tenemos que aplicar la siguiente fórmula:

$$Desviación\ del\ rendimiento\ total = \left(\frac{\eta_{real}}{\eta_{teórico}} - 1 \right) * 100$$

El **ratio de consumo** es el cociente entre la potencia real medida en la bomba y el caudal bombeado medido (kWh/m³):

$$ratio\ consumo = \frac{P_{real}}{Q_{medido}}$$

La **selección de bomba** indica si se debe cambiar la bomba o no en función de lo alejado que esté el punto de operación Q-H del punto BEP. Se muestra en tanto por ciento. Gráficamente, a partir de la Figura 7, representado sobre las curvas nominales de la bomba donde la línea azul se corresponde con la curva característica Q-H y la curva de color rojo es el rendimiento del sistema, teniendo en cuenta las pérdidas, en función del caudal.

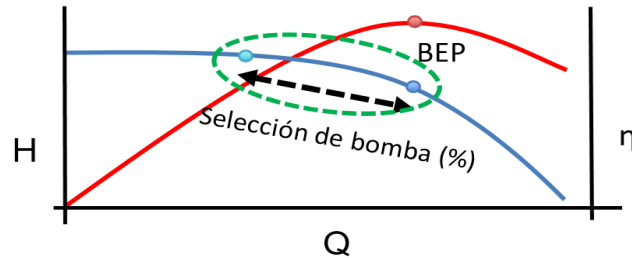


Figura 9 Representación gráfica de Selección de bomba

Por último, **el ratio de eficiencia** del sistema se corresponde al cociente entre el ratio de consumo y la altura manométrica (H) en metros de columna de agua (Wh/m³/m). La fórmula es:

$$\text{ratio eficiencia} = \frac{\text{ratio consumo}}{H} = \frac{P_{real}}{H * Q_{medido}}$$

3.3 Interpretación de los KPIs

Cada uno de estos KPI tendrá que analizarse y ver si es adecuado o no. En caso negativo se tendrán que tomar acciones al respecto. Antes de llevar a cabo cualquier acción, se debería contrastar los resultados por un especialista.

Según el indicador que esté fuera del rango deseado, se procederá a:

- Caudal actual vs caudal recomendable: ajustar el caudal de extracción para conseguir que se adecue al valor recomendable requerido. Si ambos caudales difieren más de un 10%, se tendrán que realizar ajustes.
- Caudal específico: si este indicador disminuye, supone una pérdida de eficiencia que se tendrá que recuperar. Las acciones a considerar serían realizar limpiezas debido a la posible colmatación o incrustación en filtros y rejillas o ajustar la bomba a las condiciones de un posible cambio del nivel piezométrico.
- Eficiencia hidráulica del pozo: ajuste de la bomba a las condiciones de limpieza y de desarrollo. Se procurará que esté siempre por encima de un 50%.
- Rendimiento de la bomba: posible desgaste de la bomba. Conseguir que sea lo más elevado posible.
- Desviación rendimiento total: mantenimiento en taller, ajuste. No superar un 5%.
- Selección bomba: cambio de la bomba de agua. Mantener por debajo de 10%.
- Ratio consumo unitario kWh/m³: proceder a un estudio del sistema. Inferior a 1 kWh/m³.
- Ratio de eficiencia Wh/m³/m: ajustar o sustituir la bomba. No superar un valor en torno a unos 10 Wh/m³/m.

3.4 Aplicación a casos reales

Con la parte teórica ya desarrollada, y antes de pasar a la parte experimental, se van a desarrollar estos algoritmos sobre datos reales en sondeos de dos mediciones puntuales. De este modo podemos comprobar el funcionamiento de esta algoritmia comparando con los resultados obtenidos en los diagnósticos reales.

Esta aplicación a los casos reales se desarrolla en el **ANEXO 2**.

4. Plataforma IoT

Todos los datos recogidos mediante los sensores que hemos analizado previamente deben ser tratados posteriormente con el fin de optimizar el sistema. Para recolectar los valores de las mediciones necesitaremos recurrir al uso de una plataforma capaz de leer estos sensores.

4.1 Análisis de alternativas

Necesitaremos una plataforma que permita la entrada de sensores de corriente y de tensión, que tenga buses industriales (RS485, Modbus) y con conectividad IP para hacer posible la conexión con la nube (recogida, procesamiento y envío de datos). Para empezar, vamos a analizar algunas de las opciones de plataformas que existen capaces de cumplir nuestros requerimientos:

- Raspberry Pi 3 Modelo B+
- Industrial Shields M-Duino
- Rijing Fogs7
- Ewon Flexy 205
- Conprosys Nano
- Telnet Babelgate G5001
- Siemens Simatic Iot 2020/2040
- Panasonic FP-I4C
- Avtse-Rpi-iiotg
- Tst Tscontrol
- Nexcom Nio-101

Ahora vamos a pasar a describir brevemente cada alternativa:

Raspberry Pi 3 Modelo B+



Figura 10 Raspberry Pi 3 B+

Producto de la marca Raspberry, es básicamente un pequeño ordenador. Dispone de 40 pines GPIO para entradas y salidas a los que se pueden conectar los sensores. Su sistema operativo es Raspbian de Linux. Tiene conectividad IP y opción de comunicación RS485 mediante un conversor adicional. Una desventaja importante es que no está diseñada para usarse en un entorno industrial. Su precio es de unos 30€. ^[11]

M-Duino

Se trata de un autómata de la empresa Industrial Shields basado en Arduino. Está diseñado para el entorno industrial. Se puede programar con la plataforma Arduino IDE. Dispone de puerto ethernet, RS485 y existen distintos modelos. El modelo de 13 entradas y 8 salidas digitales y analógicas configurables sería el elegido. Su precio ronda los 200€. ^[12]



Figura 11 M-Duino

Rijing Fogs7

Se trata de un PLC programable diseñado por la empresa china Rijing. Está destinado a utilizarse para aplicaciones del Internet de las cosas. Tiene pines para entradas y salidas y su sistema operativo es Linux. Utiliza el protocolo MQTT. Su precio es de unos 250€. ^[13]



Figura 12 Rijing Fogs7

Flexy 205

Es una pasarela IoT de la marca Flexy. Es modular y capaz de recoger datos y transferirlos remotamente. Es programable mediante los lenguajes Basic y Java. Permite añadir tarjetas de expansión para poder adaptarse a las necesidades requeridas. ^[14] Para nuestro caso serán necesarias:

- Tarjeta de expansión I/O card (FLX3402) para conectar todos los sensores con salida 0-10v y 4-20mA.
- Tarjeta de expansión de puerto serie dual (FLA3301) para disponer de puerto RS485.

El precio de este producto está alrededor de los 750€ y habría que sumarle el precio de los 3 módulos que se tendrán que añadir.



Figura 13 Flexy 205

Conprosys Nano

Este equipo está destinado al uso en aplicaciones IoT. Tiene entradas y salidas programables donde poder conectar los sensores, puerto RS485 y conectividad IP. Puede utilizarse con Windows o con Linux. ^[15]



Figura 14 Conprosys Nano

Babelgate G5001

Este dispositivo es una pasarela IoT de la empresa Telnet con un microprocesador programable en lenguajes como Python, Java o el sistema de programación por flujos Node-RED entre otros. Está diseñado para aplicaciones que requieran la tecnología IoT. Dispone de puerto RS485 y puertos de entrada y de salida. Es como una RaspBerry PI mejorada: más potencia, más portabilidad y más profesional. ^[16]



Figura 15 Babelgate G5001

Siemens Simatic IoT 2020/2040

Es una pasarela inteligente IoT de la empresa Siemens. Utiliza el sistema operativo Linux, dispone de puertos de entradas y salidas para los sensores y RS485. Tiene conectividad IP para la nube. Está diseñado para su uso en el entorno industrial. ^[17] Existen dos modelos:

- Simatic IoT 2020
- Simatic IoT 2040

Las diferencias entre ambos son:

- El 2040 tiene el doble de memoria RAM (1GB)
- El 2020 tiene un solo puerto ethernet mientras que el 2040 dispone de 2
- La versión 2040 tiene dos puertos serie RS232/485
- El modelo 2020 cuesta 80€ frente a los 250€ del 2040



Figura 16 Simatic IoT2040

Existe la opción de añadir un módulo de entradas y salidas preparado para la programación con el software de Arduino o parecidos. Con este módulo se pueden conectar las entradas de valores industriales. El precio de este módulo es de unos 50€.

FP-I4C

Se trata de una pasarela IoT de Panasonic diseñada para conectarse, monitorizar y operar máquinas y dispositivos de forma remota. Dispone de entradas y salidas, puerto RS485 y puertos ethernet. Utiliza los protocolos MQTT y OPC UA. ^[18]



Figura 17 FP-I4C

Avtse-Rpi-liotg



Figura 18 Avtse-RPI-IOTG

Este dispositivo de la marca Avnet es una pasarela IoT basada en la RaspBerry PI 3. Dispone de Raspbian de Linux. Tiene puerto RS485, conectividad IP y tiene la posibilidad de añadir una expansión para 40 pines GPIO donde se pueden controlar los sensores. El precio de esta plataforma es de unos 135€. ^[19]

Tscontrol

Se trata de un dispositivo de la marca TST para aplicaciones IoT en el ámbito industrial. Dispone de entradas digitales analógicas, interfaz RS485-Modbus. Está preparado para trabajar en la nube. [20]

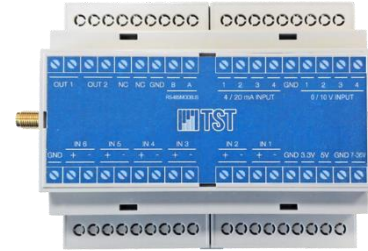


Figura 19 Tscontrol

Nio-101

Plataforma IoT de Nexcom con entradas y salidas digitales, conectividad IP, 2 puertos ethernet y un RS232/485. Dispone de sistema operativo Yocto Linux, está preparado para trabajar en entorno industrial y se le pueden añadir módulos Wi-Fi. [21]



Figura 20 Nio-101

- Tabla resumen de las alternativas:

PLATAFORMA	ENTORNO INDUSTRIAL	SISTEMA OPERATIVO	PRECIO
<u>Raspberry Pi 3 B+</u>	No	Raspbian (Linux)	30 €
<u>M-Duino</u>	Si	Linux / Android	205 €
<u>Rijing Fogs7</u>	Si	Linux	250 €
<u>Flexy 205</u>	Si	Linux	750 €
<u>Conprosys Nano</u>	Si	Windows / Linux	200 €
<u>Babelgate G5001</u>	Si	Linux	-
<u>Simatic IoT2040</u>	Si	Yocto (Linux)	250 €
<u>FP-I4C</u>	Si	Windows	450€
<u>Avtse-Rpi-iiotg</u>	Si	Raspbian (Linux)	135 €
<u>Tscontrol</u>	Si	Linux	-
<u>Nio-101</u>	Si	Yocto (Linux)	-

4.2 Elección de la plataforma

En nuestro caso vamos a trabajar con la pasarela inteligente **Siemens Simatic IoT2040**, entre otras cosas porque es una marca de confianza y ya se disponía de esta plataforma en el laboratorio. Esta opción nos permite llevar a cabo perfectamente la tarea requerida ya que permite conectar todos los sensores a ella, programarla de forma sencilla y enviar los datos a la nube. No hace falta añadirle ninguna tarjeta de expansión. Es una solución eficiente, flexible y robusta, preparado para un funcionamiento continuo las 24 horas. Además, es relativamente económica si la comparamos con el resto de las alternativas preparadas para el entorno industrial. Su temperatura de funcionamiento va desde 0°C hasta 50°C.

Se ha preferido utilizar el modelo IoT2040 frente al IoT2020 ya que sus características son mejores, teniendo por ejemplo puerto RS485 y 2 puertos ethernet, lo que puede ser una gran ventaja si fuera necesario conectarse a la pasarela sin cortar la conexión que tenía.

Tiene un rango de tensión de alimentación entre 9 y 36 VDC y un consumo máximo de corriente de 1'4 A. Esta plataforma posee un microprocesador Intel Quark x1020 y una memoria RAM de 1GB. Tiene ranura para introducir una tarjeta micro-SD y dispone de sistema operativo Yocto Linux, de código abierto. Permite la programación con lenguajes de alto nivel como Python, C++, Java, o la herramienta de desarrollo Node-RED.

Debido a la necesidad de conectar sensores con salida digital o analógica de 4-20 mA y de 0-10 V, se requiere añadir el módulo de entradas y salidas al Simatic IoT2040.

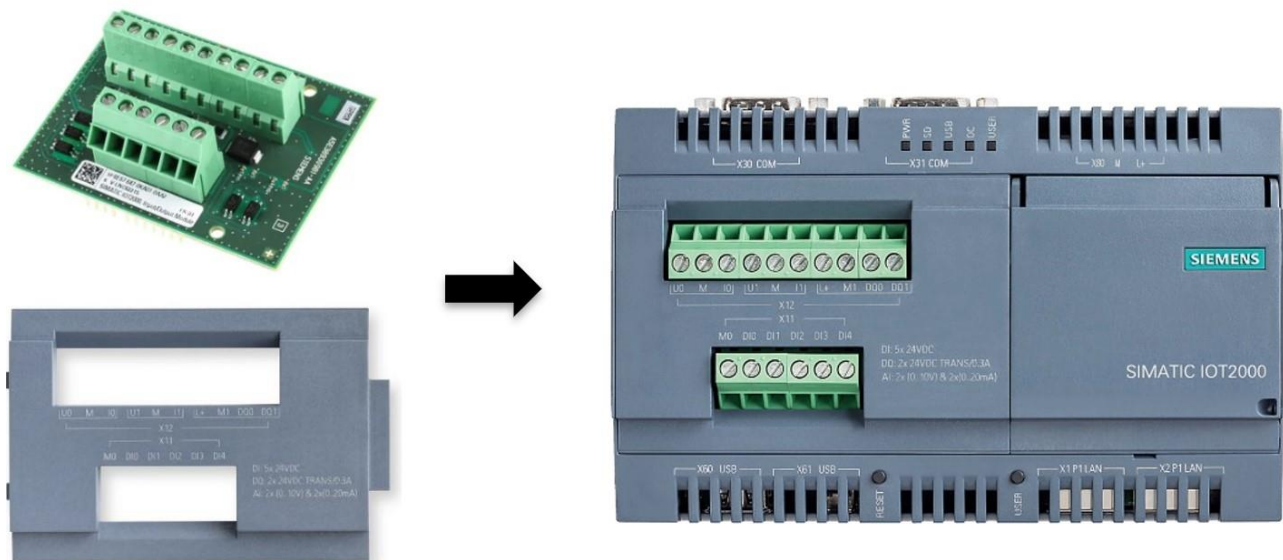


Figura 21 Modul I/O en Simatic IoT2040

Una vez elegido el dispositivo, el siguiente paso será diseñar el software. Se va a llevar a cabo un sistema de automatización al que se conectarán numerosos sensores cuyo tratamiento de los datos estará definido por el programa que se desarrolle.

4.3 Preparación del entorno

Se pasará a realizar el programa que se encargue de tratar las todas las lecturas recogidas de la instalación.

Primero tendremos que empezar configurando el IoT2040: instalación del sistema operativo, conexión al ordenador de trabajo, conexión a internet de la plataforma...

La configuración de la pasarela inteligente se desarrolla detalladamente paso por paso en el **ANEXO 3**.

Como se ha comentado anteriormente, la plataforma elegida Siemens Simatic IoT2040 tiene la posibilidad de poder programarse en diversos lenguajes de programación. Se ha optado por elegir la herramienta de programación visual **Node-RED**.

4.3.1 Node-RED

Node-RED es una herramienta de desarrollo basada en flujo y consiste en una programación visual mediante el uso de bloques y flujos, de este modo se consigue poder programar sin tener la necesidad de escribir código en algún lenguaje. Fue desarrollada por la empresa IBM en el año 2013 para conectar dispositivos de hardware, API (interfaz de programación de aplicaciones) y servicios en línea como parte del internet de las cosas.



Figura 22 Logo de Node-RED

Su propio nombre indica que se basa en nodos, que pueden ser de inyección o de función. Los nodos de inyección generan un mensaje sin requerir una entrada y lanzan el mensaje al siguiente nodo al que esté conectado. Los nodos de función en cambio tienen una entrada, la cual trabajarán y generarán una salida.^[22] Con esta arquitectura de bloques se consigue una programación visual y sencilla. Los nodos están programados en JavaScript. Se permite crear nuevos nodos para las funciones que necesitemos, programando siempre su funcionamiento en ese lenguaje.

4.4 Flujos de lectura de datos de sensores

Una vez teniendo la configuración del dispositivo establecida, el siguiente paso será hacer el flujo que sea capaz de recibir y tratar la información procedente de los distintos sensores.

Para desarrollar el flujo primero se tiene que instalar el paquete con los nodos necesarios para la toma de datos de las entradas del módulo de entradas y salidas. Como el paquete no se encuentra en el repositorio, será necesario instalarlo manualmente. Se explica cómo instalar este paquete en el Siemens Simatic IoT2040, así como el resto de los necesarios en el **ANEXO 4**.

La arquitectura para el flujo de la toma de datos de los sensores es:



Para esto primero necesitamos conocer cómo funcionan los sensores de los que disponemos: la tensión de alimentación de cada uno, el tipo de salida que proporcionan, etc.

Pese a haber elegido previamente los sensores a utilizar, para llevar a cabo este proyecto, tendremos que adaptarnos a los disponibles en el laboratorio. Esto supone investigar cada uno individualmente para conocerlos y adaptarnos a ellos.

Sensor de pulsos

Para la medida del **caudal** se dispone de un sensor óptico de pulsos digital de la marca Open Energy Monitor. Éste se alimenta a 5 VDC y proporciona una salida de datos TTL.



Figura 23 Sensor de pulsos

Usaremos una entrada digital. El flujo del programa para este sensor tiene la siguiente forma:



Figura 24 Flujo de Node-RED para medir el caudal

Se desarrolla el circuito adecuado, programa y pruebas para la lectura de este sensor, así como del resto de sensores expuestos a continuación, en el **ANEXO 5**.

Piezómetro

Para medir el **nivel de agua** (y poder obtener el descenso), tenemos un tubo piezométrico ABSD-802, que proporciona una salida de corriente entre 4 y 20 mA con un fondo de escala de 3 metros. Este sensor posee un rango de alimentación entre 12 y 36V, por lo que se conectará a los 19V que alimentan la pasarela IoT.



Figura 25 Piezómetro

Teniendo que utilizar una entrada de corriente analógica, tenemos el siguiente flujo en Node-RED:



Figura 26 Flujo de Node-RED para la medición del nivel de agua

A la lectura del nivel de agua se le resta el nivel estático para obtener el resultado del descenso.

Analizador de redes

En cuanto a la lectura de los **datos eléctricos**, se va a utilizar un analizador de redes. Se alimenta a 220V y se comunica mediante el protocolo Modbus RS485 mediante dos patillas A+ y B-.



Figura 27 Analizador de redes

De aquí se obtienen mediante Modbus las lecturas de los valores de tensión, corriente, potencia y factor de potencia. El flujo en el programa tiene la siguiente forma:

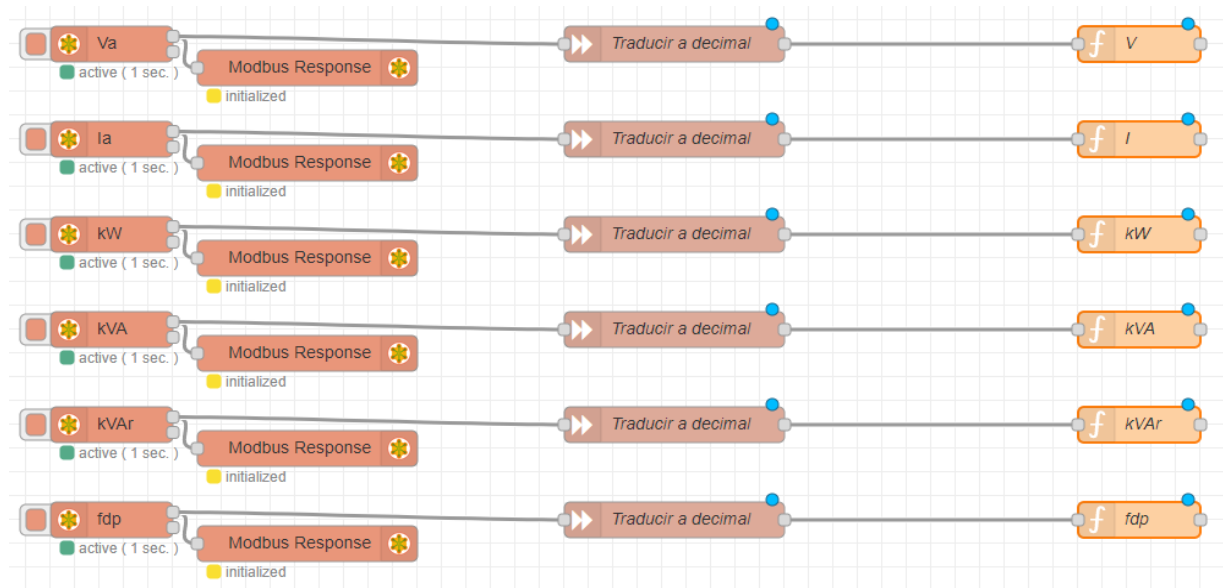


Figura 28 Flujo de Node-RED para la medición de datos eléctricos

Sensor de presión

La medición de la **altura manométrica** del agua se realiza mediante la lectura de la presión del agua una vez que esta llega a la superficie. Se dispone de un transductor de presión G1/4, que tiene como salida una tensión en el rango de 0'5 a 4'5V. Se alimenta a los 5V de la salida del regulador de tensión.



Figura 29 Sensor de presión G1/4

Utiliza una entrada de tensión analógica de la plataforma y los datos se leen con un flujo sencillo:

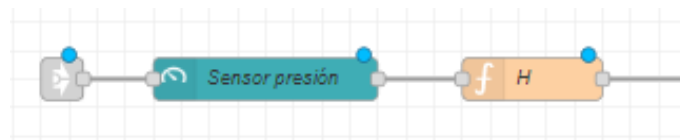


Figura 30 Flujo de Node-RED para la lectura del sensor de presión

Sensor de temperatura

La **temperatura del motor** se va a medir mediante un termistor NTC. Se alimenta a 5V e irá conectado a una entrada analógica de corriente, que variará según la temperatura.

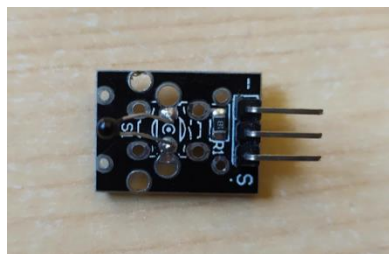


Figura 31 Termistor NTC

El flujo de lectura del termistor en Node-RED es simple y tiene la siguiente forma:



Figura 32 Flujo de Node-RED para la lectura del termistor

La conexión de los sensores al IoT2040, la configuración del programa en Node-RED de cada uno de ellos y todas las pruebas realizadas están explicados detalladamente en el **ANEXO 5**.

Finalmente, la parte del programa que corresponde a los flujos de lectura de datos de los sensores va a estar organizada como se muestra a continuación. Mediante la ayuda de *subflows*, agrupamos nodos para tener un programa mejor estructurado:

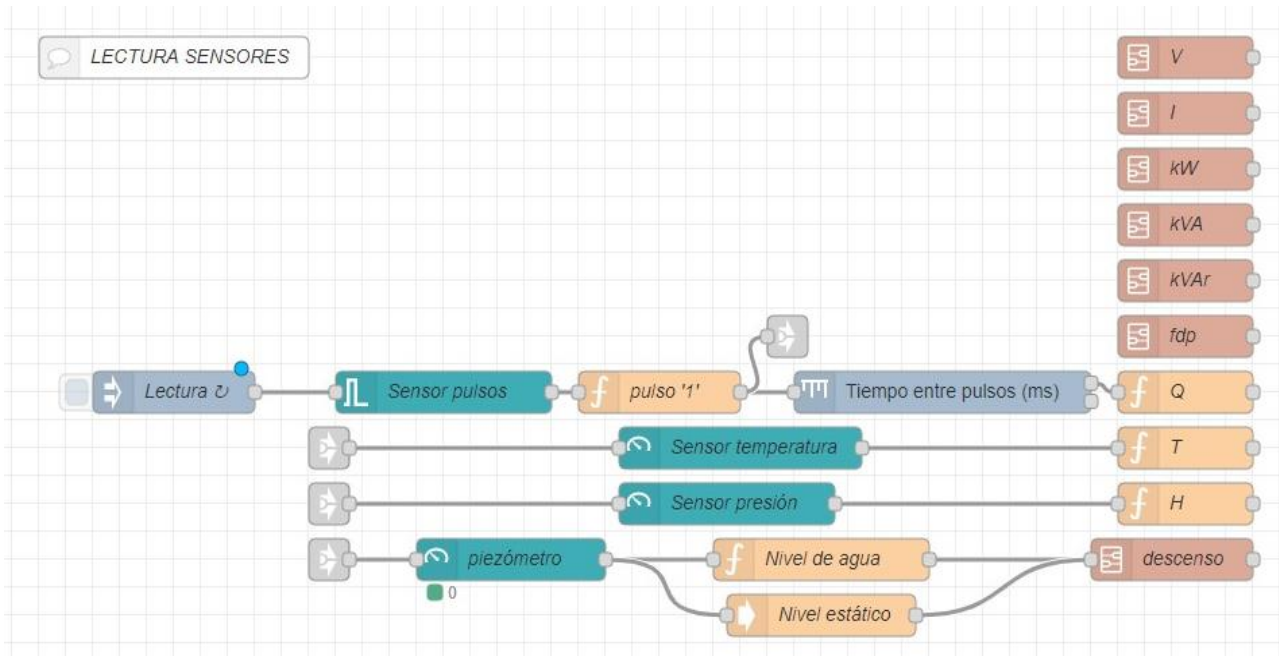


Figura 33 Flujo de Node-RED para los sensores

4.4.1 Esquemático en *Circuit Maker*

Se realiza el circuito electrónico diseñado en el programa *Circuit Maker*, software de *Altium* destinado al diseño de circuitos electrónicos. Mediante sus librerías, podemos construir el esquemático con los elementos utilizados. También permite crear componentes propios, utilidad muy interesante que sirve en este caso para añadir la plataforma IoT utilizada, ya que no se encontraba en ninguna librería.

El diseño de una manera esquemática y clara va a facilitar posibles errores en el montaje e instalación de sensores con la plataforma.

En este caso, situamos el IoT2040 en la parte central del circuito, conectándose con los sensores a través de los pines adecuados.

El esquemático de todos los componentes utilizados tiene la siguiente forma:

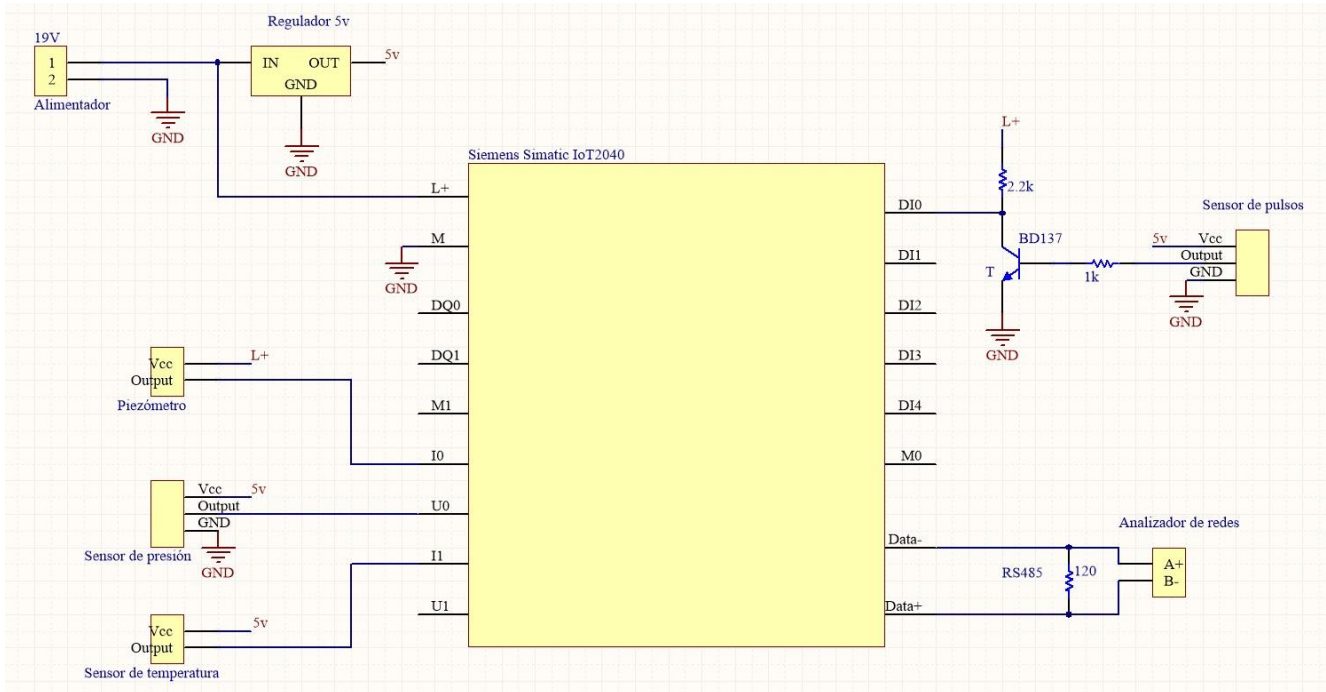


Figura 34 Esquemático del circuito

4.5 Flujos de recogida de datos de la web

Llegados al punto de poder recoger la información de los sensores, se podría requerir obtener de manera adicional los parámetros desde una base de datos o algún otro sitio público de internet en el cual se encontrasen las lecturas de los sensores u otros parámetros necesarios.

El procedimiento será primero definir el sitio web donde se encuentre el dato requerido en cada caso mediante el nodo *http request* y leer los parámetros con la frecuencia deseada.

Después, mediante el nodo *html*, inspeccionar el código de la web y coger el elemento deseado dentro del documento html.

El flujo encargado de realizar esa recogida de datos tendrá la siguiente forma:

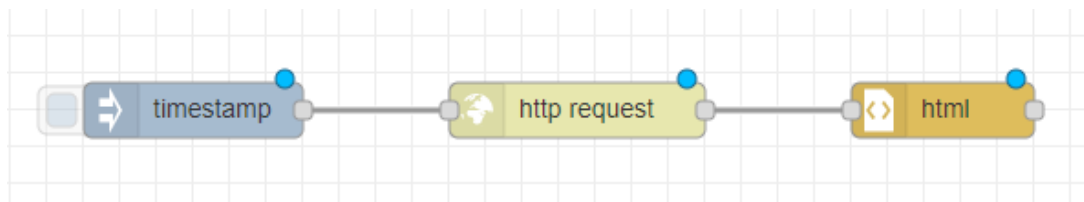


Figura 35 Flujo de Node-RED para recoger un dato desde web

Y agrupando los nodos para cada parámetro recogido:

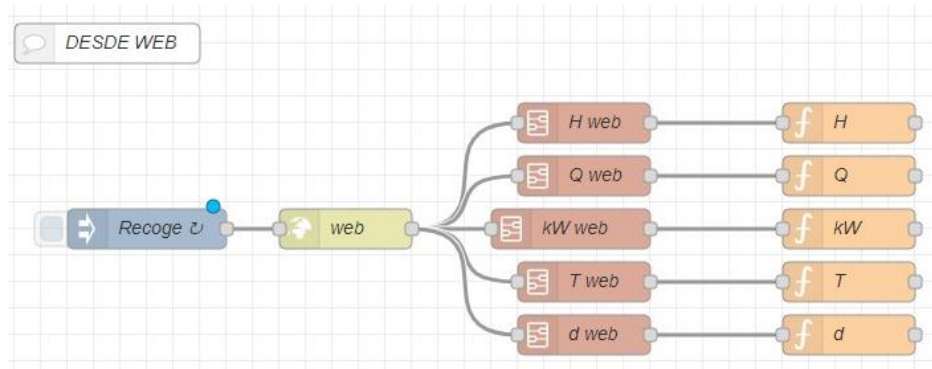
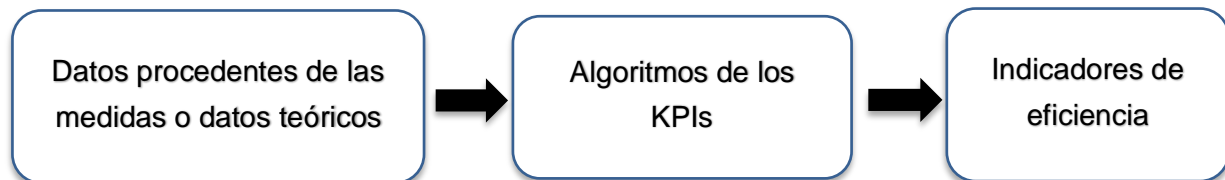


Figura 36 Flujo de Node-RED para recogida de todos los parámetros desde la web

4.6 Flujos de cálculo de indicadores

A partir de los valores procedentes de los sensores, y ya con las unidades correspondientes de cada uno, el siguiente paso será construir mediante los bloques necesarios la algoritmia del sistema que permita sacar los distintos KPIs para analizarlos posteriormente.

Con los parámetros teóricos y los reales sacados previamente, calculamos los indicadores de eficiencia:



Organizamos el programa por bloques. Se tienen que elaborar las ecuaciones que definen cada indicador dentro de los nodos del programa y trabajar con los datos.

Cuando un dato de entrada llegue a la plataforma, este se va a almacenar en una variable del *flow* (mediante `flow.set('variable', valor)`). En la función de cada KPI, se llamarán a las variables necesarias para operar con ellas (con `flow.get('variable')`).

El flujo del programa es sencillo, en los nodos de salida se encuentra la algoritmia de cada KPI, y al salir ya están listos para ser analizados. El flujo del cálculo de los KPI tendrá una frecuencia determinada, dependerá también de la frecuencia de los datos de entrada. Se eligen 5 minutos. Llegados a este punto, se tienen ya todos los parámetros del acuífero a analizar y sus indicadores de eficiencia deseados.

Tras desarrollar lo anterior, la parte de algoritmia del programa tiene la siguiente forma:

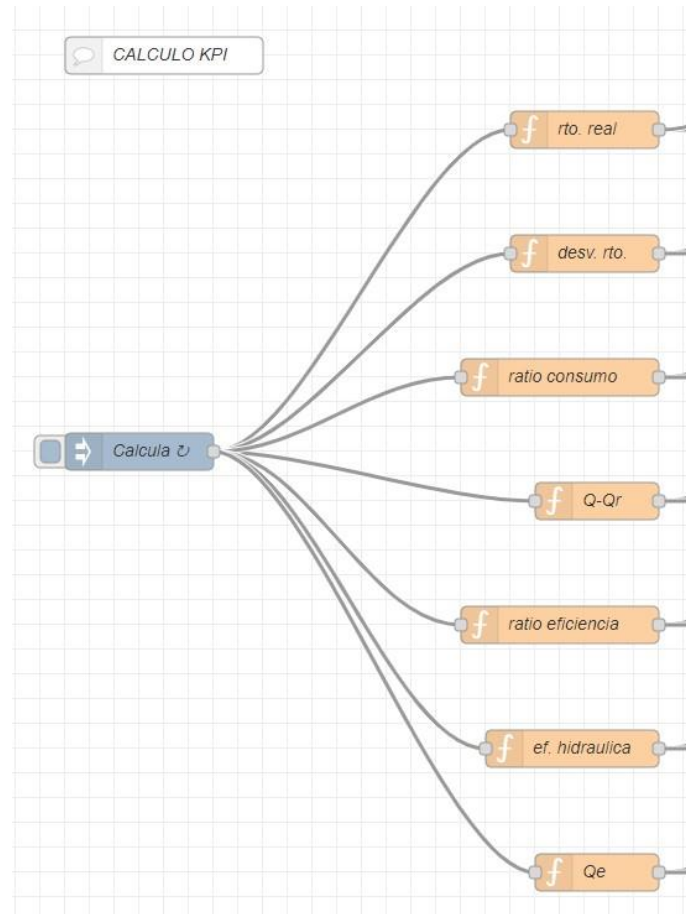


Figura 37 Flujo de Node-RED para el cálculo de los KPIs

Se añadirá también un nodo que cuente las horas de funcionamiento de la instalación, también desarrollado en el **ANEXO 5**.

El código de cada nodo del programa para el cálculo de los KPIs se expone en el **ANEXO 6**.

4.7 Notificación por correo electrónico de parámetros o KPIs fuera de rango

Una vez hechas las gráficas y con todos los datos accesibles, va a ser muy interesante también que se avise cuando hay un valor de una medición o un KPI fuera del rango establecido como normal. Esto puede suponer un ahorro de tiempo para la toma de alguna acción, con el fin de corregir o prevenir las posibles ineficiencias que surgiesen en la instalación de bombeo.

Para llevar a cabo este aviso, se va a programar para cada dato de salida una función que detecte los valores fuera de rango, y posteriormente, enviarlos a un nodo que se encargue de enviarlos vía correo electrónico.

El flujo de Node-RED tendrá la siguiente forma para cada KPI o parámetro (se muestra el ejemplo para el caudal específico):

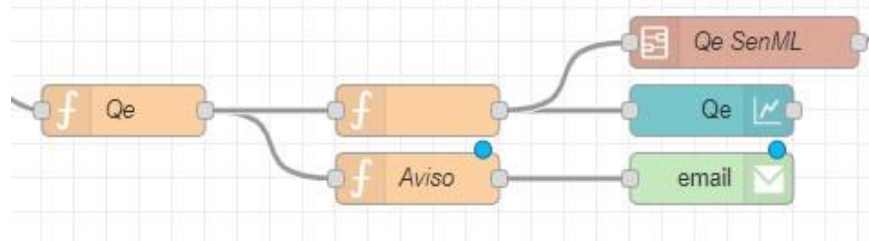


Figura 38 Flujo de envío de datos fuera de rango vía e-mail

Finalmente, comprobamos que cuando un dato sale del rango considerado como correcto, se notifica automáticamente mediante un e-mail (a la dirección de correo electrónico que se haya establecido):

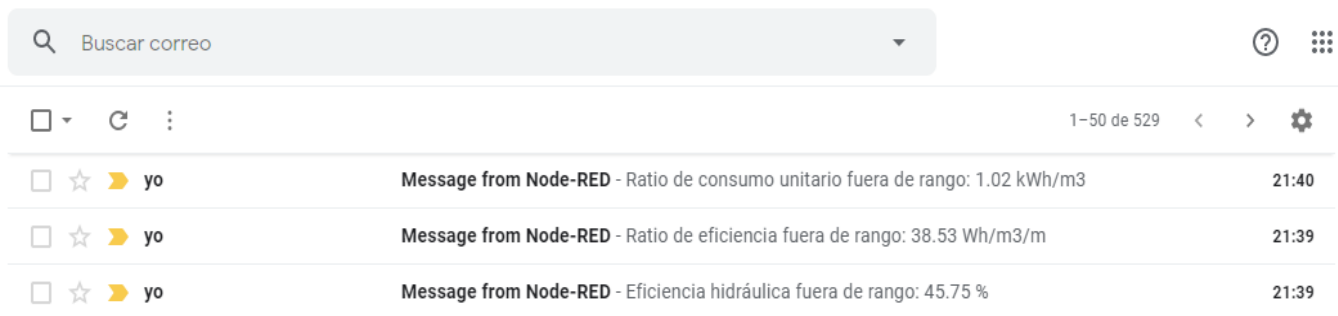


Figura 39 Bandeja de entrada del correo electrónico con mensajes de aviso

5. Pruebas en entorno real

5.1 Descripción del escenario de pruebas

Una vez se ha llegado al punto en el cual se han desarrollado en la plataforma los algoritmos, aparece la posibilidad de obtener los datos de una instalación real para poder implementarlos. La instalación pertenece a la empresa *La Zaragozana*, de la cual se podrá recibir información en tiempo real de una instalación de bombeo de agua (nivel piezométrico, volumen bombeado, potencia...). Esta será la fuente de entrada de datos de nuestro programa.

Se tiene acceso a información como el nivel estático del pozo, que es de 10'6 metros, o la altura manométrica, que es de 32 m.c.a. Además, se han obtenido los siguientes resultados en un ensayo de bombeo escalonado a diferentes caudales, permitiendo obtener los valores de A, B y n para poder sacar el descenso teórico:

ENSAYO BOMBEO ESCALONADO								
Q l/s	Q m3/día	d real	A	B	n	d cálculo	d teórico	Eficiencia
6	518,4	0,71	1,10E-03	5,16E-07	2	0,71	0,57	80,4%
16	1382,4	2,51	1,10E-03	5,16E-07	2	2,51	1,52	60,7%
26	2246,4	12,8	1,10E-03	5,16E-07	2	5,07	2,47	48,7%

Figura 40 Ensayo de bombeo escalonado en la instalación real

$$d_{real} = A * Q_{medido} + B * Q_{medido}^n \quad d_{teórico} = A * Q_{medido}$$

La bomba de la instalación de la cual recogeremos la información es una Grundfos SP-95-10, con motor MMS8000 de 45 kW. Su curva Q-H es la siguiente:

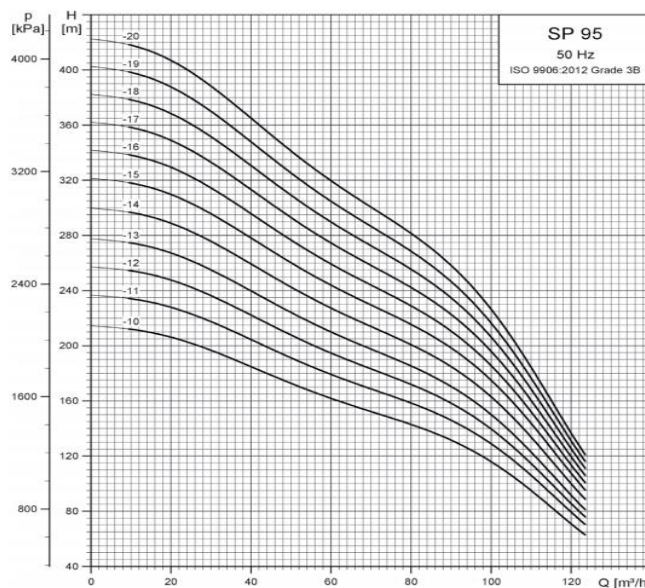


Figura 41 Curva Q-H de la bomba

5.1.1 Primera prueba puntual

El primer paso antes de empezar a recoger los datos de manera continua será aplicar la algoritmia desarrollada previamente a alguna medición puntual del sistema real. Esto se va a realizar mediante una hoja de cálculo en Excel, en el cual se introducen las fórmulas y los resultados de las mediciones.

Primero recogemos las mediciones en una tabla:

DATOS MEDIDOS			
Medición	1	2	3
Q (l/s)	15,3454545	12,631884	15,99444444
Nivel dinámico (m)	14,0706009	13,583305	13,994
kW	11,06	12,89	13,89

Figura 42 Mediciones en pozo real

Y se pasa a sacar los valores teóricos y los indicadores clave:

DATOS TEÓRICOS			
Medición	1	2	3
rto. teórico	75%	75%	75%
d teórico	1,46E+00	1,20E+00	1,52E+00
kW teóricos	4,81888194	3,9667484	5,022682074

$$d_{teórico} = A * Q_{medido}$$

$$P_{teórica} = \frac{0.736 * Q * H}{75}$$

KPI's			
Medición	1	2	3
d (m)	3,48060086	2,9933047	3,404
Qe (l/s/m)	4,40885214	4,2200461	4,6987205
Desviación rendimiento (%)	-41,88%	-58,96%	-51,78%
Eficiencia hidráulica	61,7%	66,1%	60,7%
Ratio consumo (kWh/m3)	0,20011825	0,2833683	0,2411942
Ratio eficiencia (Wh/m3/m)	6,25369541	8,8552581	7,5373176
Rendimiento real (%)	43,59%	30,78%	36,17%

$$descenso = Nivel\ dinámico - Nivel\ estático$$

$$Q_e = \frac{Q_{medido}}{d}$$

$$Desviación\ del\ rendimiento\ total = \left(\frac{\eta_{real}}{\eta_{teórico}} - 1 \right) * 100$$

$$Eficiencia\ hidráulica = \frac{d_{teórico}}{d_{real}} * 100$$

$$ratio\ consumo = \frac{P_{real}}{Q_{medido}}$$

$$ratio\ eficiencia = \frac{ratio\ consumo}{H} = \frac{P_{real}}{H * Q_{medido}}$$

Figura 43 Datos teóricos y KPIs de mediciones reales

5.2 Sensorización y recogida de datos

El siguiente paso va a ser acceder a la información de los sensores de la instalación y establecer una comunicación. Después, se podrá pasar a tratar esos datos.

La información disponible se puede descargar en CSV o recoger mediante la utilización del protocolo **MQTT**. MQTT (Message Queing Telemetry Transport) es un protocolo de comunicación *machine to machine* muy popularizado en el campo del IoT, debido entre otras cosas a su sencillez y reducido consumo de ancho de banda. Como base para la comunicación generalmente se ejecuta sobre TCP/IP, pero cualquier protocolo de red que proporcione conexiones bidireccionales ordenadas y sin pérdidas puede admitirlo. [23]

Su arquitectura se basa en un *broker* que gestiona la red, al cual los clientes mandan los paquetes con la información y esperan respuesta, como se muestra el siguiente esquema:

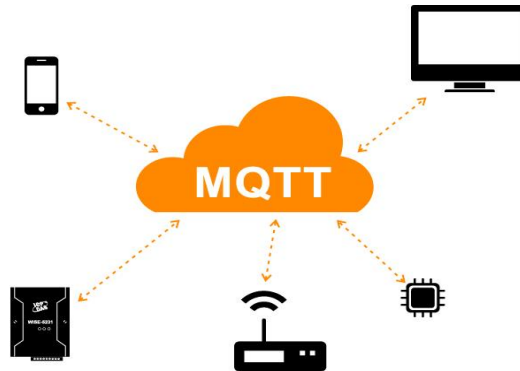


Figura 44 Arquitectura del protocolo MQTT

Los broker MQTT filtran los mensajes que reciben para discriminar a qué clientes suscritos es entregado, este filtro se denomina *topic*. Los clientes publican el mensaje indicando un *topic*, y si alguna suscripción pasa ese filtro, se le transmite el mensaje. [24]

Los datos en tiempo real a los que se tiene acceso y que serán nuestra fuente de entrada del programa son los siguientes:

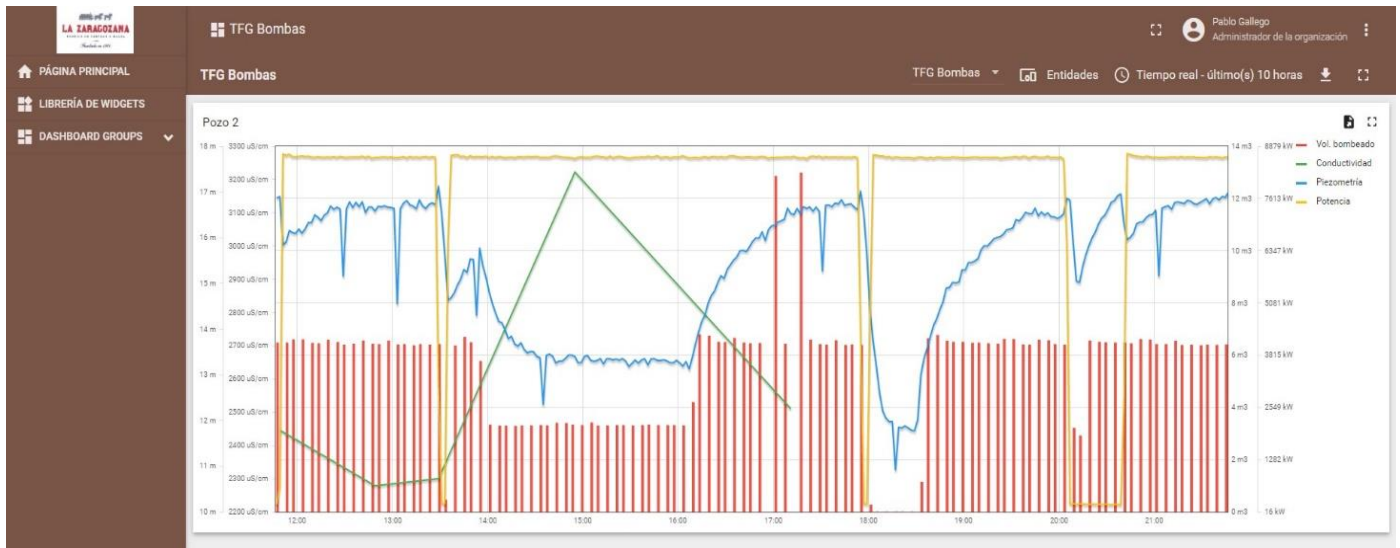


Figura 45 Datos de la instalación real

Antes de recoger los datos directamente, se comprueba la información exportando los últimos datos del sistema en formato CSV. Se programa y comprueba en Node-RED un flujo capaz de recibir los datos del fichero como entradas para la algoritmia. El flujo es el siguiente (en los nodos de las funciones se accede a la posición en cada caso del dato requerido):

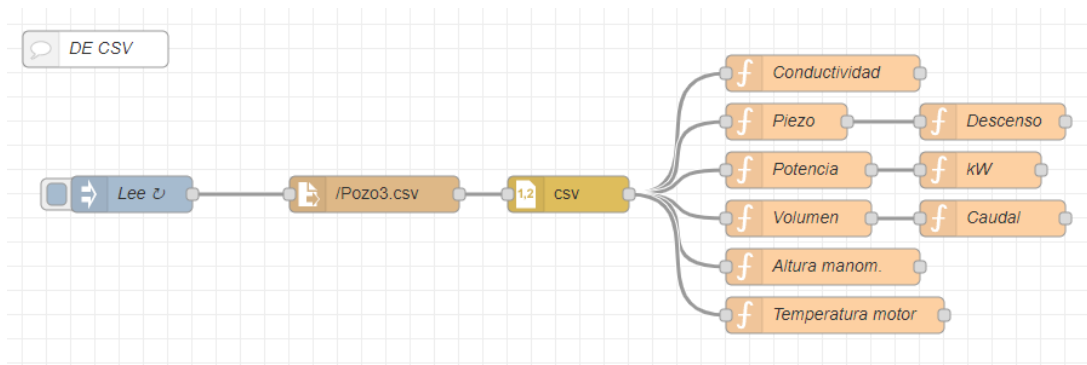


Figura 46 Flujo para recoger datos de un fichero CSV

El siguiente paso va a ser recoger directamente los datos en tiempo real de la instalación mediante el protocolo de comunicación MQTT. El acceso a los datos se realiza mediante el servidor *senializar.unizar.es* y el topic *bombas/zaragozana/#*. Los datos entran al programa como objeto JSON (un formato de texto para el intercambio de datos, es un subconjunto de la notación literal de objetos de JavaScript.) y en formato SenML (Sensor Measurement Lists) ^[24], con la siguiente forma:

```
"[{"bn":"urn:dev:mac:a840410001818ca5/","t":"1591700009445","n":"ACI1","v":19.78},{n":"ACI2","v":20.41},{n":"piezometro","v":14.07}]"
```

Mediante el programa, se tiene que extraer la información necesaria para tratar. El nodo rodeado recoge los datos vía MQTT. El flujo de recogida de estos datos tiene la siguiente forma:

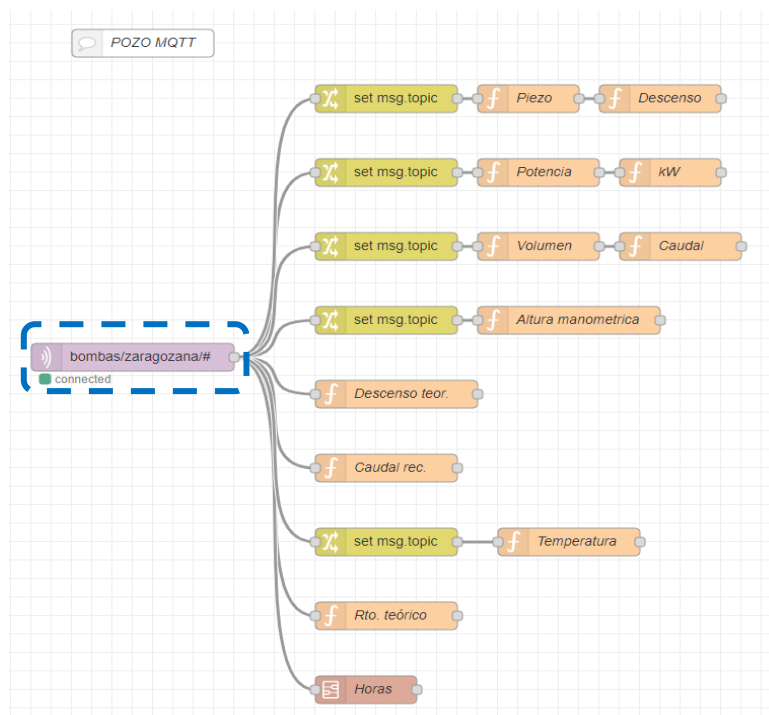


Figura 47 Flujo de recogida de datos de instalación real vía MQTT

Estos datos son los que definirán los valores de entrada de la algoritmia de los KPIs. El código del flujo anterior se encuentra en el **ANEXO 6**.

5.3 Monitorización y envío a base de datos

5.3.1 Flujos de interfaz gráfica local

Ya con los parámetros listos y los KPIs calculados, se va a pasar a enviar toda la información a una interfaz gráfica local donde podrá visualizarse.

Node-RED dispone de una *dashboard* donde se pueden monitorizar las salidas de los nodos del programa. Vamos a utilizar esta herramienta para nuestros parámetros a controlar. Se realiza mediante los nodos instalados para mandar los datos al tablero. En este caso se utiliza el nodo correspondiente a la representación en gráficas (indicados en línea discontinua en la siguiente imagen en los flujos de KPIs y de parámetros de la instalación).

Con todo esto, se procede a hacer una interfaz gráfica destinada a la lectura de los indicadores y los parámetros más importantes junto con información relevante de una manera más visual. Va a estar diseñada para la revisión y el control de la instalación en tiempo real, tipo SCADA.

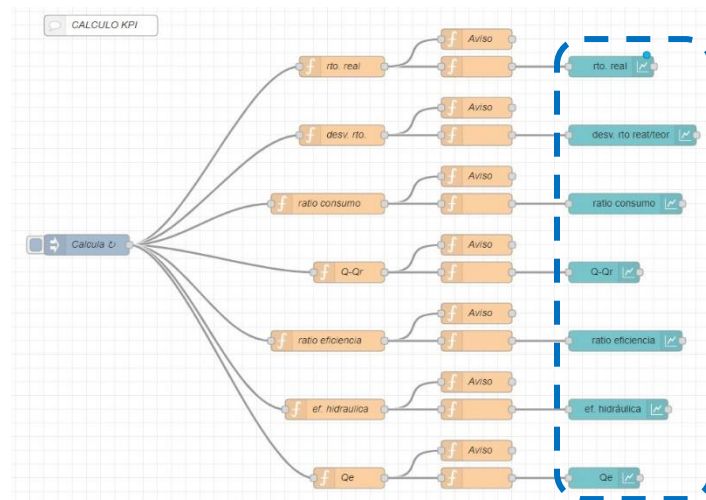


Figura 48 Flujo de Node-RED para representar los parámetros en la dashboard

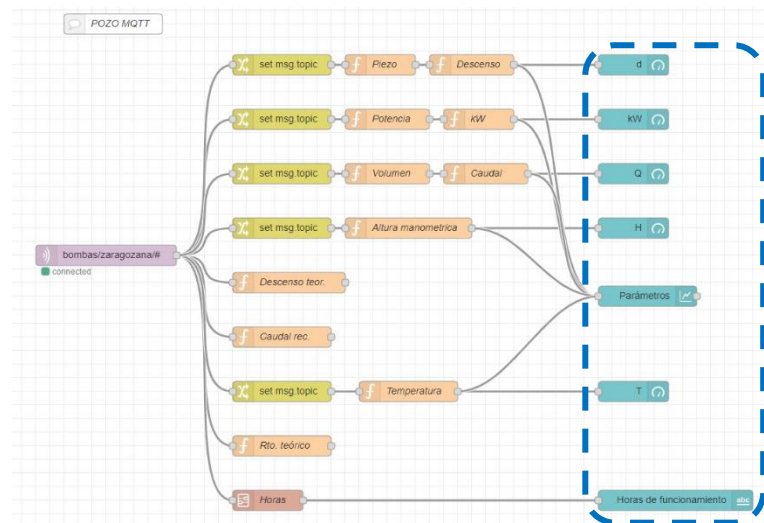


Figura 49 Flujo de Node-RED para representar los KPIs en la dashboard

La *dashboard* de Node-RED tiene muchas limitaciones a la hora del diseño de su interfaz, no está permitido mover libremente los elementos que aparecen en ella. Realizamos el diseño de la imagen que aparecerá en la interfaz mediante el software de diseño gráfico *Adobe Illustrator* intentando representar la instalación real de una manera esquemática.

Se muestran los resultados en tiempo real en tres interfaces entre las cuales se puede navegar:

- **Parámetros en tiempo real:** Aquí se van a representar los parámetros actuales de la instalación.
- **KPIs en gráficas en tiempo real:** Se representa la evolución con el tiempo de los indicadores de eficiencia junto a su explicación.
- **Histórico de la evolución de los parámetros:** Muestra la evolución de los parámetros de control con el tiempo. Se añade también las horas de funcionamiento totales de la instalación.

Las pantallas de la interfaz gráfica local quedarán finalmente de la siguiente manera:

- Parámetros en tiempo real:

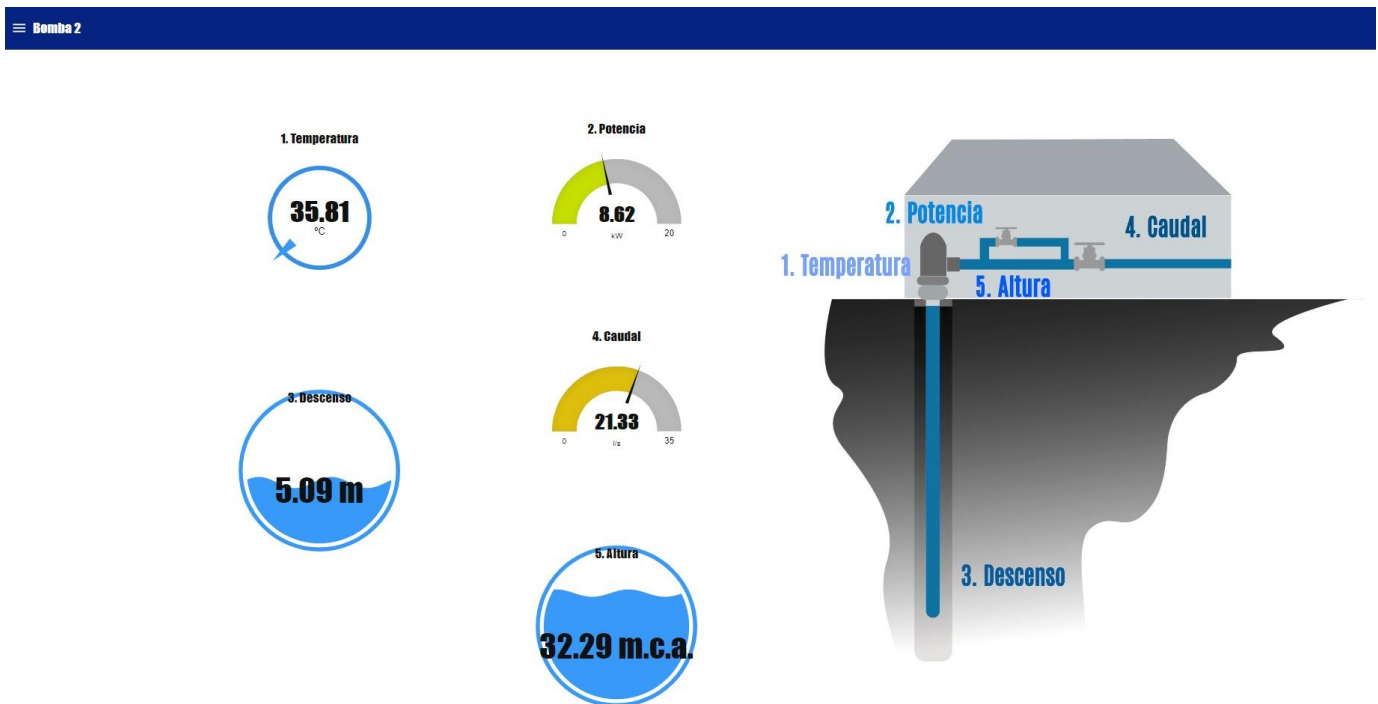


Figura 50 Interfaz gráfica de parámetros en tiempo real

- KPIs en gráficas en tiempo real:



Figura 51 Interfaz gráfica de KPIs en tiempo real

- Histórico de la evolución de los parámetros:

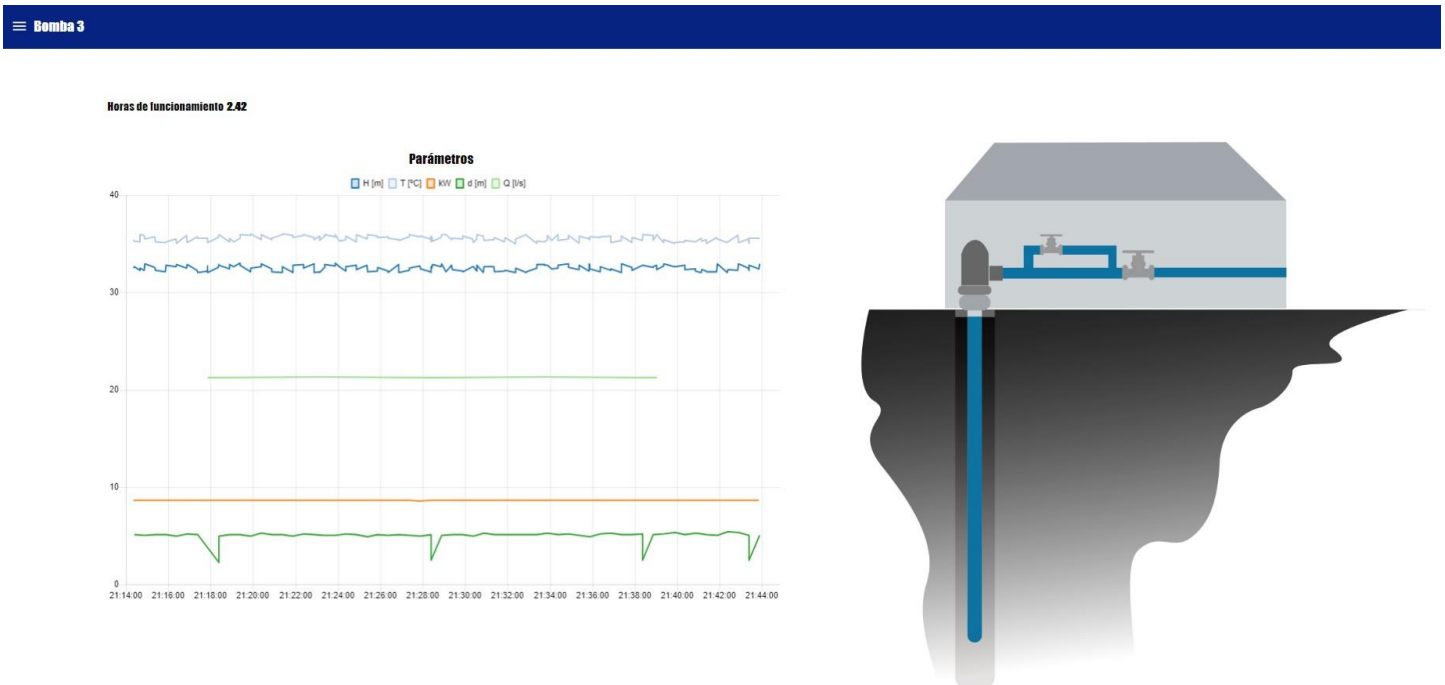


Figura 52 Interfaz gráfica de la evolución de los parámetros en el tiempo

5.3.2 Flujos de conexión MQTT

El último paso va a ser enviar toda la información obtenida previamente a la nube. Los datos se subirán vía MQTT. Así, el control de la instalación se podrá realizar desde cualquier lugar con conexión a internet.

En este caso, la información se va a subir a una base de datos InfluxDB, que permite almacenar datos de series de tiempo y es muy utilizada para este tipo de aplicaciones de monitorización para el IoT.

Los mensajes que contienen la información que se va a subir estarán en formato SenML. Para darle este formato a los datos antes de enviarlos, se tendrán que convertir a un String JSON. El código para la conversión se incluye en el **ANEXO 6**. Los datos quedarán de la siguiente forma:

```
"[{\"bn\": \"urn:dev:org:bomba/\", \"n\": \"descenso\", \"u\": \"m\", \"v\": 10.32}]"
```

Con los datos ya listos para enviar, se va a recurrir al nodo del programa existente para el envío de mensajes con protocolo MQTT. A este nodo llegan todos los parámetros e indicadores. Para que envíe correctamente los datos tendrá que configurarse con los parámetros adecuados.

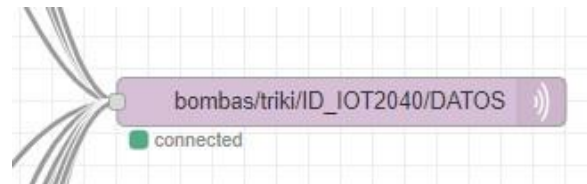


Figura 53 Nodo de envío MQTT de Node-RED

La información se enviará a un broker MQTT localizado en el servidor *senializar.unizar.es* con el puerto 4883 al *topic bombas/ triki /ID_IOT2040/DATOS*. Mediante un procesado de los datos transmitidos, estos se almacenan en la base de datos.

Para comprobar que los datos se están almacenando de forma correcta, se accede a una instancia de Chronograf instalada en el mismo servidor (Chronograf es un software de visualización y monitorización de datos para base de datos InfluxDB, y forma parte del paquete InfluxData).



Figura 54 Monitorización en base de datos influxDB

Se puede acceder y ver todos los datos enviados a lo largo del tiempo. Además, desde allí se pueden elegir los datos que monitorizar y comprobar que se suben de manera adecuada.

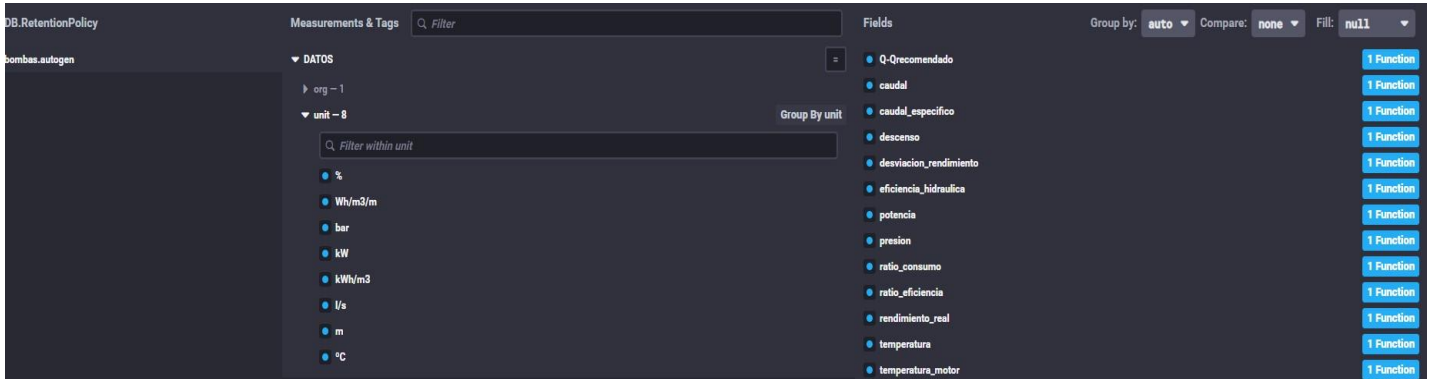


Figura 55 Interfaz de la base de datos y monitorización

6. Conclusiones y trabajo futuro

En este trabajo se ha diseñado y desarrollado un sistema basado en el IoT capaz de llevar el control de una instalación de bombeo de agua, una aplicación muy interesante debido a los ahorros que podrían aparecer. Se tiene un control en tiempo real desde cualquier lugar de todos los parámetros que informan sobre posibles desgastes a ineficiencias. Se ha desarrollado tanto la parte de hardware como la de software, con la utilización de la tecnología IoT.

Si evaluamos los resultados obtenidos, podemos concluir que se han cumplido los objetivos del trabajo, siendo capaces de recoger, tratar y monitorizar a través de internet los datos deseados de la instalación.

Este trabajo desarrollado con el fin de trabajar en una instalación de bombeo, podría implementarse en cualquier otro campo. Al tener una plataforma IoT conectada a internet capaz de trabajar en un entorno industrial de forma continua, puede aplicarse para el control y mantenimiento predictivo de otro tipo de instalaciones, pudiendo monitorizar las variables o indicadores que se requiriesen según el caso de aplicación. Bastaría con sustituir, si fuese necesario, por los sensores adecuados para la toma de distintos parámetros, dependiendo del tipo de instalación.

En un futuro, probablemente la mayoría de empresas cuenten con sistemas similares al desarrollado en este trabajo, ya que es una manera de aprovechar el imparable avance de la tecnología y el internet, estando más presente en nuestras vidas, y, por lo tanto, en muchas instalaciones. La inversión en este tipo de sistemas de control puede suponer el ahorro de grandes cantidades de dinero en algunos casos.

6.1 Posible instalación del sistema

Para colocar la plataforma y sensores listos para funcionar en un caso real, hay que empezar teniendo en cuenta que debe tener conexión a internet. Se trata de controlar una instalación de agua subterránea, donde los elementos de conectividad estarán en el exterior. Muy probablemente se encuentre fuera de un entorno urbano, esto significa que seguramente no tengamos una conexión de internet por cable.

Para solucionar este problema, se ha pensado en la instalación de un router 3G/4G, a través de la red móvil, con una tarjeta SIM de un operador con cobertura en la zona.

Es posible que, en un entorno rural, fuera del núcleo de población, la cobertura de la red móvil pueda ser escasa. Para solucionar este problema, buscaríamos un router con interfaz wan por red móvil 3G/4G, y que se le pudiera conectar una antena externa y separada del router mediante un extensor de cable coaxial. De esta manera, se instalaría el router junto al resto de elementos del sistema, y la antena en un punto con señal móvil. Se tendría que instalar una antena de alta ganancia.

6.2 Conclusiones personales

La realización de este trabajo ha supuesto para mí una gran adquisición de conocimientos que han hecho que valga la pena todo el tiempo y esfuerzo invertidos en este proyecto.

Me ha servido para aprender sobre el uso de la tecnología del internet de las cosas (IoT). Era un tema que me llamaba la atención desde hace algún tiempo y que veo con mucho futuro, siendo una tecnología con la que me gustaría seguir trabajando y aprendiendo. No existe ninguna asignatura en el grado donde se trabajen estas tecnologías de comunicaciones.

Empecé el trabajo con mucha curiosidad y algo de inquietud debido a que nunca había trabajado con ninguna plataforma similar a la que iba a utilizar en este proyecto: Siemens Simatic IoT2040. Gracias a esto, he aprendido a programar y trabajar con una plataforma más industrial a lo que estaba acostumbrado. Además, he aprendido sobre la importancia de las nuevas tecnologías como fuentes de ahorro para muchas instalaciones.

La oportunidad de poder programar esta plataforma inteligente en lenguaje Node-RED me ha parecido muy interesante, ya que conocía este lenguaje y me parecía novedoso. Nunca he trabajado con una programación mediante el uso de flujos. Además, me ha servido para aprender del protocolo de comunicación MQTT y algo de programación en JavaScript, que tampoco había tenido la posibilidad de hacer previamente.

Este trabajo me ha hecho poner en práctica lo aprendido a lo largo de estos años estudiando ingeniería. Me ha gustado trabajar con herramientas nuevas y diferentes que me han servido para aprender a buscar soluciones de un modo más práctico y real.

7. Bibliografía

- [1] Cantudo Muñoz, Ángel. (2019). SUEZ: *Uso eficaz de indicadores en la gestión de aguas subterráneas*, Madrid.
- [2] Sánchez, F. Javier. Universidad de Salamanca, departamento de geología: *Eficiencia de una captación*. Disponible online en: <http://hidrologia.usal.es/temas/Eficiencia.pdf>
- [3] Blog Máquinas eléctricas: *Pérdidas*. Disponible online en: <https://maquinaselectricasblog.wordpress.com/perdidas/>
- [4] Hispanagua, Sistema español de información sobre el agua: *Hidrogeología*. Disponible online en: <https://hispaagua.cedex.es/datos/hidrogeologia>
- [5] Ingemecánica. Ingeniería, consultoría y formación: *Tutorial 206: Cálculo de instalaciones de bombeo de agua*. Disponible online en: <https://ingemecanica.com/tutorialsemanal/tutorialn206.html>
- [6] Díaz Ortiz, J. Ernesto. (2006). *Mecánica de los fluidos e hidráulica*. Disponible online en: <https://books.google.es/books?id=3jHhWwwQqp4C&printsec=frontcover&hl=es#v=onepage&q&f=false>
- [7] SUEZ: Smart Water. *Proyectos para el sector del agua, S.A.U.* Disponible online en: <https://www.suez.es/-/media/suez-es/files/pdf-spain/suez-smart-water-presentacion-comercial.pdf>
- [8] Tecfluid: *Caudalímetros electromagnéticos*. Disponible online en: <https://tecfluid.com/caudalimetros-electromagneticos-en-linea-o-insercion/>
- [9] Hidroconta: *WI contador Woltmann*. Disponible online en: <http://files.riego-agricola.com/200000316-03ab304a37/WI-CONTADOR%20WOLTMANN.pdf>
- [10] (2019). Enviraio: *Tipos de sensores de temperatura y principales diferencias*. Disponible online en: <https://enviraio.es/tipos-de-sensores-de-temperatura/>
- [11] Datasheet RASPBERRY PI 3 MODELO B+: https://components101.com/sites/default/files/component_datasheet/Raspberry%20Pi%203%20Datasheet.pdf
- [12] Datasheet INDUSTRIAL SHIELDS M-DUINO: <https://www.robotshop.com/media/files/content/i/ind/pdf/m-duino-plc-arduino-ethernet-19r-i-os-relay---analog---digital-plus-datasheet.pdf>
- [13] Datasheet RIJING FOGS7: <https://spanish.alibaba.com/product-detail/rijing-fogs7-wifi-programmable-plc-iot-gateway-fog-sever-60814850048.html>

- [14] Datasheet EWON FLEXY 205:
<https://www.ewon.biz/docs/librariesprovider10/ewon-english/datasheets/datasheet-flexy-205.pdf>
- [15] Datasheet CONPROSYS NANO:
https://www.contec.com/api/downloadlogger?download=-/media/contec/product/datasheet/en/ds_cpsnmcb271s1041_en.pdf
- [16] Datasheet TELNET BABELGATE G5001: https://www.telnet-ri.es/wp-content/uploads/2018/02/DS_IoT_BabelGate_G5001_ES.pdf
- [17] Datasheet SIEMENS SIMATIC IOT 2020/2040:
https://cache.industry.siemens.com/dl/files/658/109741658/att_899623/v1/iot2000_operating_instructions_e_en-US.pdf
- [18] Datasheet PANASONIC FP-I4C: https://www.panasonic-electric-works.com/cps/rde/xbcr/pew_eu_en/mn_fpi4c_quickstartguide_peweu_en.pdf
- [19] Datasheet AVTSE-RPI-IIOTG: <https://www.farnell.com/datasheets/2828937.pdf>
- [20] Datasheet TST TSCONTROL: <http://www.tst-sistemas.es/Docs/TSconTrol-GPRS.pdf>
- [21] Datasheet NEXCOM NIO-101: <https://www.nexcom.com/fildata/getpdf/46a719cf-9251-49cd-9104-0a33251d2178>
- [22] Darío, Jesús. Toptal: *Programación visual con Node-Red: Conectando el IoT con facilidad*. Disponible online en: <https://www.toptal.com/nodejs/programacion-visual-con-node-red-conectando-el-internet-de-las-cosas-con-facilidad>
- [23] Llamas, Luis. (2019). Blog Luis Llamas: *¿Qué es MQTT? Su importancia como protocolo IoT*. Disponible online en: <https://www.luisllamas.es/que-es-mqtt-su-importancia-como-protocolo-iot/>
- [24] Llamas, Luis. (2019). Blog Luis Llamas: *Qué son y cómo usar los topics en MQTT correctamente*. Disponible online en: <https://www.luisllamas.es/que-son-y-como-usar-los-topics-en-mqtt-correctamente/>
- [25] Universidad de Bremen TZI. (2018). Standard SennML: *Sensor Measurement Lists (SenML)*. Disponible online en: <https://tools.ietf.org/html/rfc8428>

8. Anexos

8.1 ANEXO 1: Elección de sensores y justificación

Tras analizar los parámetros necesarios a tomar, debemos buscar qué sensores se adecúan a las características y requerimientos de nuestro sistema y realizar su elección.

Nivel de agua:

Debido al caso que se aborda, necesitaremos una sonda que pueda llegar a medir niveles de decenas de metros, trabajar en un ambiente húmedo y a un rango de temperaturas del agua en torno a 10 °C. Se ha escogido la sonda de nivel LS-10 de la marca Wika, ya que es perfecta para la medición de nivel de agua en acuíferos, es robusto y llega a un límite de profundidad de hasta 100 metros. Tiene una precisión del 0'5% y salida industrial (4 - 20 mA). Puede comprarse en su página web por un precio en torno a 400€.



Caudal:

El rango de medición requerido es bastante elevado: de 0 a 100 l/s. Se elige el caudalímetro PCD-TFM ya que tiene una precisión muy elevada, es robusto y mide sin problemas caudales en el rango que abordamos, que es bastante elevado. Tiene unos parámetros de funcionamiento adecuados y proporciona una salida analógica entre 4 y 20 mA.

Se ha optado por usar un caudalímetro de turbina ya que tienen una eficiencia muy alta, son duraderos y de los más utilizados en la industria de proceso.



Presión:

Para obtener este parámetro utilizaremos el transductor de presión G1/4 ya que es idóneo para trabajar con agua. Es resistente y de acero inoxidable. Proporciona una salida de tensión proporcional a la lectura y puede medir presiones en el rango que se requiere en este caso. Su precio ronda los 20€ en diversas páginas web.



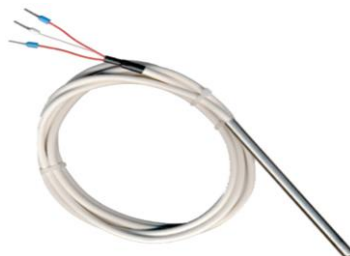
Datos eléctricos:

Elegimos el analizador de redes de Walfront para poder tener de salida los valores de tensión, corriente, potencia activa, reactiva, aparente y factor de potencia. Es económico, su precio ronda los 40€ en Amazon.



Temperatura:

Vamos a elegir una sonda RTD Pt100 ya que es el sensor más común, muy estable y preciso. Perfecto para usarse en las condiciones de temperatura requeridas en este caso. Al ser muy común, puede encontrarse fácilmente en cualquier tienda desde unos 10€.



8.2 ANEXO 2: Aplicación de algoritmia a casos reales

En este anexo se va a desarrollar la algoritmia expuesta de manera teórica sobre los datos obtenidos en dos casos de mediciones reales.

Sondeo Villamera

Este sondeo abastece a un municipio de 1400 habitantes, por lo que es crítico. El gasto eléctrico era muy elevado y la cantidad de agua extraída era cada vez menor. Se realizó un diagnóstico de eficiencia, para el cual se obtuvieron las siguientes medidas:

HIDRÁULICOS	
Q:	4,5 l/s
NE:	7,44 m
ND:	8,04 m
P:	7 b

ELÉCTRICOS S/C		ELÉCTRICOS C/C	
V:	425 V	V:	425 V
I:	46 A	I:	39,5 A
KW:	27,1 kW	KW:	27,1 kW
KVA:	34,1 kva	KVA:	28,3 kva
KVAr	20,6 kvar	KVAr	7,91 kvar
cosΦ	0,80	cosΦ	0,96

ENSAYO BOMBEO ESCALONADO CORTA DURACIÓN (1 h / escalón)							
Stage	Q (l/s)	d real (m)	B coef	C coef	n coef	d (calculo) (m)	Eff (%)
1	1,5	0,15	1,05E-03	7,57E-07	2,1	0,16	86,8
2	2,5	0,30	1,05E-03	7,57E-07	2,1	0,29	78,9
3	3,5	0,45	1,05E-03	7,57E-07	2,1	0,44	72,1
4	4,5	0,60	1,05E-03	7,57E-07	2,1	0,62	66,3

A partir de las medidas anteriores, obtuvieron los siguientes indicadores de eficiencia con su correspondiente diagnóstico:

INDICADORES (Acuífero – Pozo)		DIAGNÓSTICO	INDICADORES (Instalación Electromecánica)		DIAGNÓSTICO
Q:	4,5 l/s	NO ADECUADO	Q:	4,5 l/s	
d:	0,60 m	BAJO	Ht:	79,44 mca	
Qe:	7,5 l/s/m	MEDIO ALTO	$\eta(r)$ (%)	12,9%	NO ADECUADO
Ecuación Pozo:	$d: 1,049E-3Q + 7,572E-07Q^{2,1}$	ADECUADO	$\eta(t)$ (%)	50,0%	
Eficiencia Hidráulica:	66%	ADECUADA	Dva $\eta(r)$ Vs $\eta(t)$	-74%	NO ADECUADO
Trasmisividad	850 m ² /día	MUY ALTA	Ajuste Bomba:	N/D %	N/D
			Selección Bomba:	N/D %	N/D
			ratio Consumo:	1,673 kWh/m ³	NO ADECUADO
			ratio Eficiencia:	21,058 wh/m ³ /m	NO ADECUADO
			Comp. Reactiva	SI	ADECUADO

Para demostrar que nuestra algoritmia es correcta se va a probar a generar las tablas con los datos e indicadores y comprobar si tenemos los mismos resultados. Creamos una plantilla mediante la herramienta Microsoft Excel e introducimos los datos:

DATOS MEDIDOS		Q (m3/h)	Q (m3/día)
Q (l/s)	4,5	16,2	388,8
Nivel Estático (m)	7,44		
Nivel dinámico (m)	8,04		
P (bar)	7		
kW	27,1		
H (m.c.a.)	79,44		

DATOS TEÓRICOS	
rto. teórico	50%
d teórico	0,41
kW teóricos	3,5

KPI's	
d (m)	0,6
Qe (l/s/m)	7,5
Eficiencia hidráulica	66,2%
Rendimiento real	12,9%
Desviación rendimiento	-74%
Ratio consumo (kWh/m3)	1,67283951
Ratio eficiencia (Wh/m3/m)	21,0578991

$$\text{descenso} = \text{Nivel dinámico} - \text{Nivel estático}$$

$$Q_e = \frac{Q_{\text{medido}}}{d} \quad \text{Eficiencia hidráulica} = \frac{d_{\text{teórico}}}{d_{\text{real}}} * 100$$

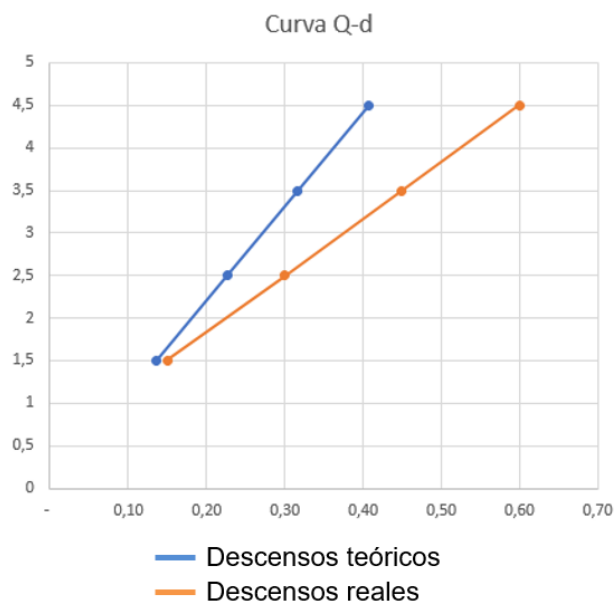
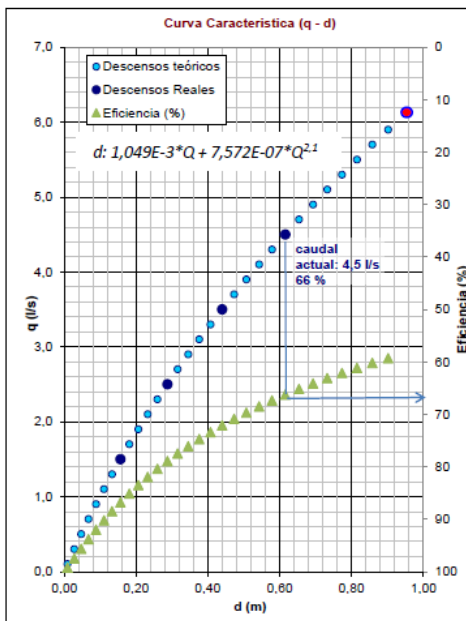
$$\eta_{\text{real}} = \frac{P_{\text{teórica}}}{P_{\text{real}}} * 100 \quad \text{Desviación del rendimiento total} = \left(\frac{\eta_{\text{real}}}{\eta_{\text{teórico}}} - 1 \right) * 100$$

$$\text{ratio consumo} = \frac{P_{\text{real}}}{Q_{\text{medido}}} \quad \text{ratio eficiencia} = \frac{\text{ratio consumo}}{H} = \frac{P_{\text{real}}}{H * Q_{\text{medido}}}$$

ENSAYO BOMBEO ESCALONADO									
Q l/s	Q m3/día	d real	A	B	n	d cálculo	d teórico	Eficiencia	
1,5	129,6	0,15	1,05E-03	7,57E-07	2,1	0,16	0,14	86,8%	
2,5	216	0,3	1,05E-03	7,57E-07	2,1	0,29	0,23	78,9%	
3,5	302,4	0,45	1,05E-03	7,57E-07	2,1	0,44	0,32	72,1%	
4,5	388,8	0,6	1,05E-03	7,57E-07	2,1	0,62	0,41	66,2%	

Vemos que los KPIs obtenidos mediante nuestros algoritmos coinciden con los del caso real.

Comparamos de paso las curvas Q-d:



Ahora comprobaremos también su actualización al cambiar la bomba en la instalación real:

INDICADORES (Acuífero – Pozo)		DIAGNÓSTICO	INDICADORES (Instalación Electromecánica)		DIAGNÓSTICO
Caudal Actual:	7,6 l/s	ADECUADO	Q:	7,6 l/s	ADECUADO
d:	1,40 m	BAJO	Ht:	84,4 mca	ADECUADO
Qe:	5,4 l/s/m	MEDIO ALTO	$\eta_t(r)$ (%)	50,8%	ADECUADO
Ecuación Pozo:	$d: 1,070E-3Q + 4,052E-07Q^2$	ADECUADO	$\eta_t(t)$ (%)	51,2%	ADECUADO
Eficiencia Hidráulica:	52 %	ADECUADA	Dva $\eta_t(r)$ Vs $\eta_t(t)$	-1%	ADECUADO
Transmisividad	903 m ² /día	MUY ALTA	Ajuste Bomba:	< 5 %	ADECUADO
			Selección Bomba:	4 %	ADECUADO
			ratio Consumo:	0,453 kWh/m ³	ADECUADO
			ratio Eficiencia:	5,367 wh/m ³ /m	ADECUADO
			Comp. Reactiva	SI	ADECUADO

Y con nuestra plantilla vuelven a coincidir los resultados:

DATOS MEDIDOS		Q (m ³ /h)	Q (m ³ /día)
Q (l/s)	7,6	27,36	656,64
Descenso (m)	1,4		
P (bar)	-		
kW	12,39		
H (m.c.a.)	84,4		

DATOS TEÓRICOS	
rto. teórico	51%
d teórico	0,70
kW teóricos	6,3

KPI's	
d (m)	1,4
Qe (l/s/m)	5,42857143
Eficiencia hidráulica	52,4%
Rendimiento real	50,8%
Desviación rendimiento	-1%
Ratio consumo (kWh/m ³)	0,45285088
Ratio eficiencia (Wh/m ³ /m)	5,36553172

$descenso = Nivel\ dinámico - Nivel\ estático$

$$Q_e = \frac{Q_{medido}}{d} \quad Eficiencia\ hidráulica = \frac{d_{teórico}}{d_{real}} * 100$$

$$\eta_{real} = \frac{P_{teórica}}{P_{real}} * 100 \quad Desviación\ del\ rendimiento\ total = \left(\frac{\eta_{real}}{\eta_{teórico}} - 1 \right) * 100$$

$$ratio\ consumo = \frac{P_{real}}{Q_{medido}} \quad ratio\ eficiencia = \frac{ratio\ consumo}{H} = \frac{P_{real}}{H * Q_{medido}}$$

ENSAYO BOMBEO ESCALONADO								
Q l/s	Q m ³ /día	d real	A	B	n	d cálculo	d teórico	Eficiencia
7,6	656,64	1,4	1,07E-03	4,05E-07	2,2	1,34	0,70	52,4%

Sondeo Catogrí

Sondeo perteneciente a un conjunto de sondeos de abastecimiento de una comarca de 25000 habitantes. Se produjo un incremento del precio de la extracción de agua debido a una bajada en el caudal. Los datos de indicadores que se tomaron para el estudio fueron:

INDICADORES (Acuífero – Pozo)		DIAGNÓSTICO
Q:	23,8 l/s	NO ADECUADO
d:	18,08 m	NO ADECUADO
Qe:	1,32 l/s/m	MUY BAJO
Ecuación Pozo:	$d: 6,00E-3Q + 4,43E-08Q^{2,5}$	NO ADECUADO
Eficiencia Hidráulica:	68 %	ADECUADA
Trasmisividad	350 m ² /día	ALTA

INDICADORES (Instalación Electromecánica)		DIAGNÓSTICO
Q:	23,8 l/s	
Ht:	20,5 mca	
$\eta_t(r)$ (%)	57 %	ADECUADO
$\eta_t(t)$ (%)	58 %	
Dva $\eta_t(r)$ Vs $\eta_t(t)$	-1%	ADECUADO
Ajuste Bomba:	< 5 %	ADECUADO
Selección Bomba:	21 %	NO ADECUADO
ratio Consumo:	0,100 kWh/m ³	ADECUADO
ratio Eficiencia:	4,9 wh/m ³ /m	ADECUADO
Comp. Reactiva	SI	ADECUADO

Comprobando con nuestra algoritmia:

DATOS MEDIDOS		Q (m ³ /h)	Q (m ³ /día)
Q (l/s)	23,8	85,68	2056,32
Nivel Estático (m)	2,46		
Nivel dinámico (m)	20,54		
P (bar)	-		
kW	10,8		
H (m.c.a.)	20,5		

DATOS TEÓRICOS	
rto. teórico	58%
d teórico	12,34
kW teóricos	6,156

KPI's	
d (m)	18,08
Qe (l/s/m)	1,31637168
Eficiencia hidráulica	59,2%
Rendimiento real	57,0%
Desviación rendimiento	-2%
Ratio consumo (kWh/m ³)	0,12605042
Ratio eficiencia (Wh/m ³ /m)	6,14880098

$$descenso = \text{Nivel dinámico} - \text{Nivel estático}$$

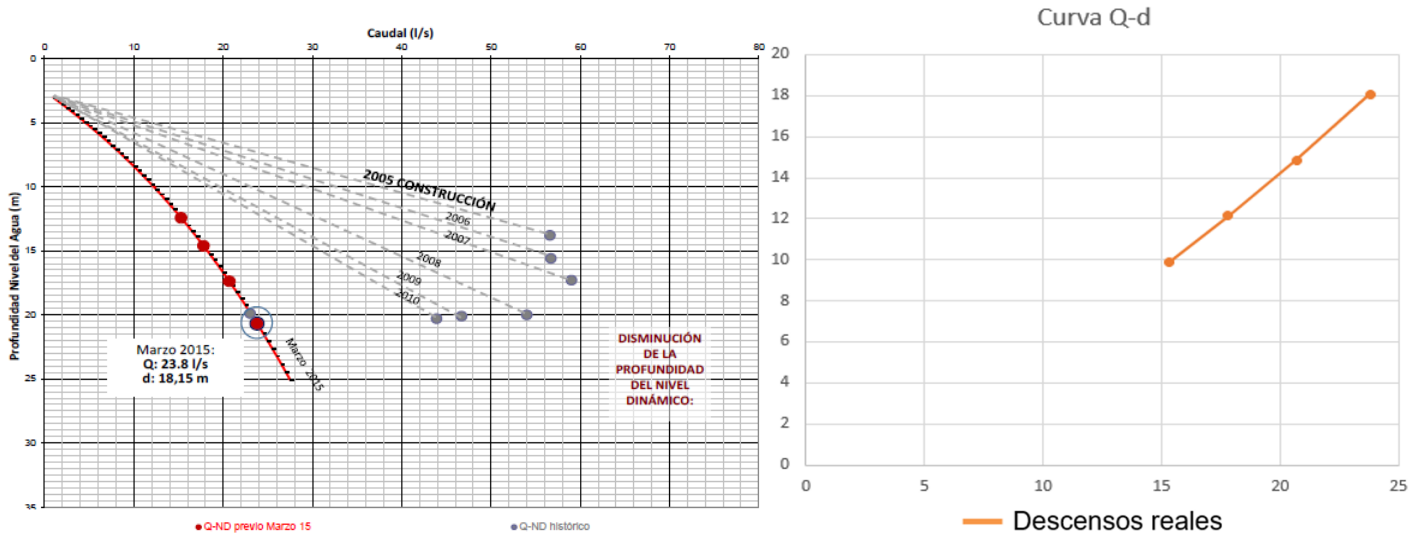
$$Q_e = \frac{Q_{medido}}{d} \quad \text{Eficiencia hidráulica} = \frac{d_{teórico}}{d_{real}} * 100$$

$$\eta_{real} = \frac{P_{teórica}}{P_{real}} * 100 \quad \text{Desviación del rendimiento total} = \left(\frac{\eta_{real}}{\eta_{teórico}} - 1 \right) * 100$$

$$\text{ratio consumo} = \frac{P_{real}}{Q_{medido}} \quad \text{ratio eficiencia} = \frac{\text{ratio consumo}}{H} = \frac{P_{real}}{H * Q_{medido}}$$

ENSAYO BOMBEO ESCALONADO								
Q l/s	Q m ³ /día	d real	A	B	n	d cálculo	d teórico	Eficiencia
15,3	1321,92	9,87	6,00E-03	4,43E-08	2,5	10,75	7,93	73,8%
17,8	1537,92	12,14	6,00E-03	4,43E-08	2,5	13,34	9,23	69,2%
20,7	1788,48	14,84	6,00E-03	4,43E-08	2,5	16,72	10,73	64,2%
23,8	2056,32	18,08	6,00E-03	4,43E-08	2,5	20,83	12,34	59,2%

Y la curva Q-d de esta medición:



Para la mejora que se desarrolló mediante una limpieza del sistema, los resultados fueron:

Escalón	Q		d TOTAL (m)	Eficiencia (%)	Qe (l/s/m)
	l/s	m3/día			
1	16,7	1443	0,5	86,3	31,5
2	26,0	2246	0,9	78,8	27,4
3	41,5	3586	1,7	67,9	23,8
4	55,6	4804	2,6	59,8	21,8
5	69,0	5962	3,6	53,5	21,8
6	76,3	6592	4,2	50,5	19,2

Caudal Explotación:	
Caudal explotación post:	76,3 l/s
Mejora:	221 %

Vol Anual	404.930 m3/año
ratio Consumo	0,080 kwh/m3
ratio Eficiencia	4,190 wh/m3/m

Y para comprobar, los valores obtenidos con nuestra algoritmia:

DATOS MEDIDOS		Q (m3/h)	Q (m3/día)
Q (l/s)	76,3	274,68	6592,32
Descenso (m)	4,2		
P (bar)	-		
kW	22		
H (m.c.a.)	19,09		

DATOS TEÓRICOS	
rto. teórico	58%
d teórico	2,14

KPI's	
d (m)	4,23
Qe (l/s/m)	18,0170053
Eficiencia hidráulica	50,4%
Ratio consumo (kWh/m3)	0,0800932
Ratio eficiencia (Wh/m3/m)	4,19555785

$$\text{descenso} = \text{Nivel dinámico} - \text{Nivel estático}$$

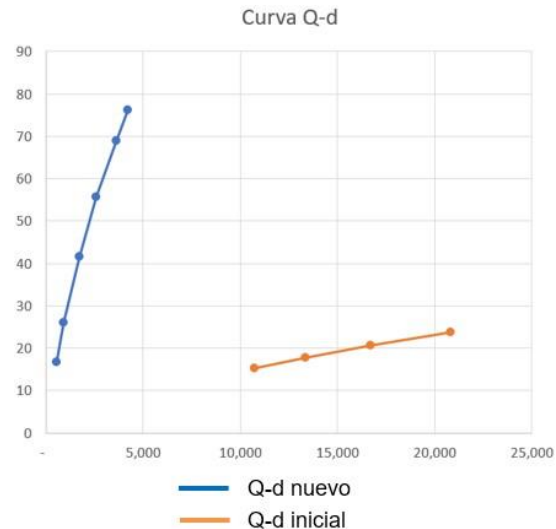
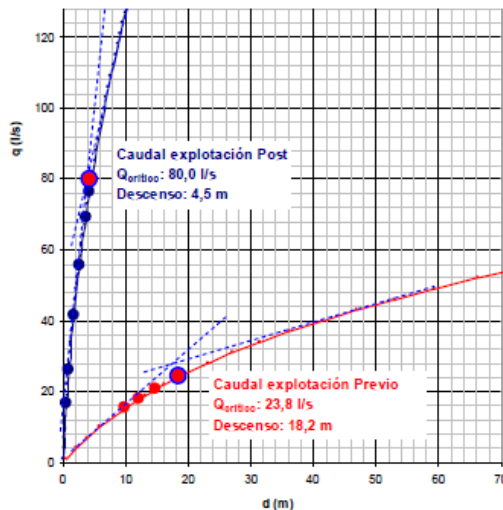
$$Q_e = \frac{Q_{\text{medido}}}{d} \quad \text{Eficiencia hidráulica} = \frac{d_{\text{teórico}}}{d_{\text{real}}} * 100$$

$$\eta_{\text{real}} = \frac{P_{\text{teórica}}}{P_{\text{real}}} * 100 \quad \text{Desviación del rendimiento total} = \left(\frac{\eta_{\text{real}}}{\eta_{\text{teórico}}} - 1 \right) * 100$$

$$\text{ratio consumo} = \frac{P_{\text{real}}}{Q_{\text{medido}}} \quad \text{ratio eficiencia} = \frac{\text{ratio consumo}}{H} = \frac{P_{\text{real}}}{H * Q_{\text{medido}}}$$

ENSAYO BOMBEO ESCALONADO									
Q l/s	Q m3/día	d real	A	B	n	d cálculo	d teórico	Eficiencia	
16,7	1442,88	0,5	3,24E-04	8,32E-09	2,2	0,54	0,47	86,3%	
26	2246,4	0,9	3,24E-04	8,32E-09	2,2	0,92	0,73	78,7%	
41,5	3585,6	1,7	3,24E-04	8,32E-09	2,2	1,71	1,16	67,9%	
55,6	4803,84	2,6	3,24E-04	8,32E-09	2,2	2,60	1,56	59,8%	
69	5961,6	3,6	3,24E-04	8,32E-09	2,2	3,61	1,93	53,4%	
76,3	6592,32	4,2	3,24E-04	8,32E-09	2,2	4,23	2,14	50,4%	

Comparando las curvas Q-d antes y después de la mejora en ambos casos tenemos:



Con esto comprobado, se puede dar por demostrado el correcto funcionamiento de los algoritmos desarrollados.

8.3 ANEXO 3: Configuración de la conexión del IoT2040

Para empezar, tenemos que configurar nuestra plataforma. Lo primero será instalar una imagen de firmware que contiene el sistema operativo Yocto Linux. Siemens nos proporciona en su página web el archivo de esta imagen con su sistema operativo y alguna otra configuración para nuestro dispositivo.

Tendremos que instalar la imagen descargada en una tarjeta micro-SD para posteriormente introducirla en la ranura del IoT2040.



Tras alimentar por primera vez nuestra plataforma (en este caso mediante una fuente de alimentación de 19V de un ordenador portátil) con la micro-SD en su interior, se tendrá que esperar a que el sistema redimensione la tarjeta.

Cuando ese proceso haya terminado, pasamos a acceder al dispositivo a través de nuestro ordenador.

La conexión inicial se establece físicamente vía interfaz Ethernet RJ45. Utilizamos para esto el puerto X1P1LAN del IoT2040. A nivel software, la conexión es mediante cliente con protocolo SSH, para lo cual se usará la herramienta "Putty" en el ordenador.

Los datos de conexión por defecto proporcionados por el fabricante son:

- Dirección IP: 192.168.200.1 / 255.255.255.0
- Usuario: *root*

A nivel de protocolo, la primera conexión al IoT2040 la haremos configurando TCP/IP del ordenador con una dirección IP del mismo rango que el dispositivo. Por ejemplo 192.168.200.200. Una vez establecida la conexión del terminal Putty con el dispositivo, ya podemos empezar a dar los primeros comandos a nuestra plataforma.

La primera tarea será configurar y conectarla a internet para poder instalar y actualizar los módulos necesarios para el proyecto. Para esto, se va a integrar tanto el IoT2040 como el ordenador en la red local que tenemos disponible, con direcciones dentro de su rango.

Datos de la red local disponible:

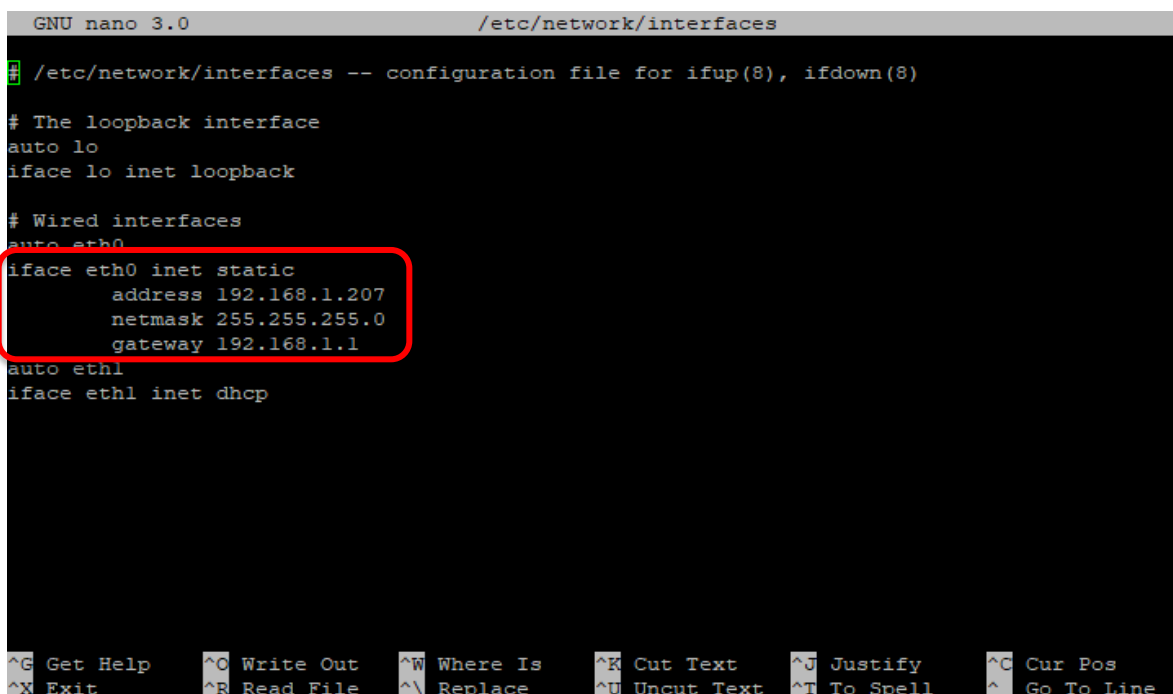
- Rango de direcciones: 192.168.1.X
- Máscara de subred: 255.255.255.0
- Puerta de enlace de salida a internet o Gateway (router): 192.168.1.1
- DNS: 80.58.0.33 (servidor de la propia operadora que nos proporciona servicio de internet)

Se decide asignar a nuestro IoT la dirección IP estática (libre en la red): 192.168.1.207.

Tras establecer la conexión SSH Putty – IoT2040 y pasar la autenticación (usuario: root), editaremos los siguientes ficheros de configuración mediante el editor de Linux “nano”.

- Accedemos al fichero /etc/network/interfaces y modificamos las siguientes líneas:

```
iface eth0 inet static
    address 192.168.1.207
    netmask 255.255.255.0
    gateway 192.168.1.1
```



```
GNU nano 3.0 /etc/network/interfaces
# /etc/network/interfaces -- configuration file for ifup(8), ifdown(8)

# The loopback interface
auto lo
iface lo inet loopback

# Wired interfaces
auto eth0
iface eth0 inet static
    address 192.168.1.207
    netmask 255.255.255.0
    gateway 192.168.1.1
auto eth1
iface eth1 inet dhcp

^G Get Help      ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify     ^C Cur Pos
^X Exit          ^R Read File    ^\ Replace      ^U Uncut Text  ^T To Spell    ^_ Go To Line
```

Así le asignamos una IP estática para que siempre tenga la misma, la elegimos dentro del rango que tenemos y con un número que esté libre, en este caso se elige 192.168.1.207.

Del mismo modo, editamos el fichero `/etc/resolv.conf` para configurar el DNS. Añadimos la línea: `nameserver 80.58.0.33` (nuestro DNS).

Salimos guardando el fichero en ambos casos. Una vez realizados los cambios, reiniciamos para que se apliquen correctamente (reboot).

Se tiene que cambiar también la dirección de nuestro ordenador por una libre de la Lan en la que vamos a integrarnos para que así estén todos los elementos en el mismo rango que la red disponible.

Ahora es el momento de conectar físicamente mediante un cable de red el dispositivo a un puerto Lan del router de internet. Se conecta el ordenador también a la misma red, ya sea por cable o por wifi.

Una vez integrados en la red local, volvemos a conectar mediante Putty al IoT2040. Tras confirmar que la conectividad es correcta, se procede a actualizar el software.

8.4 ANEXO 4: Instalación de paquetes en Node-RED

Para el desarrollo del programa en Node-RED va a ser necesaria la instalación de unos paquetes de nodos que no trae inicialmente. Estos van a ser:

- Nodos para la lectura de los puertos del módulo de entradas y salidas
- Nodo para contar intervalos de tiempo
- Nodo para realizar operaciones matemáticas
- Nodo para enviar correos electrónicos

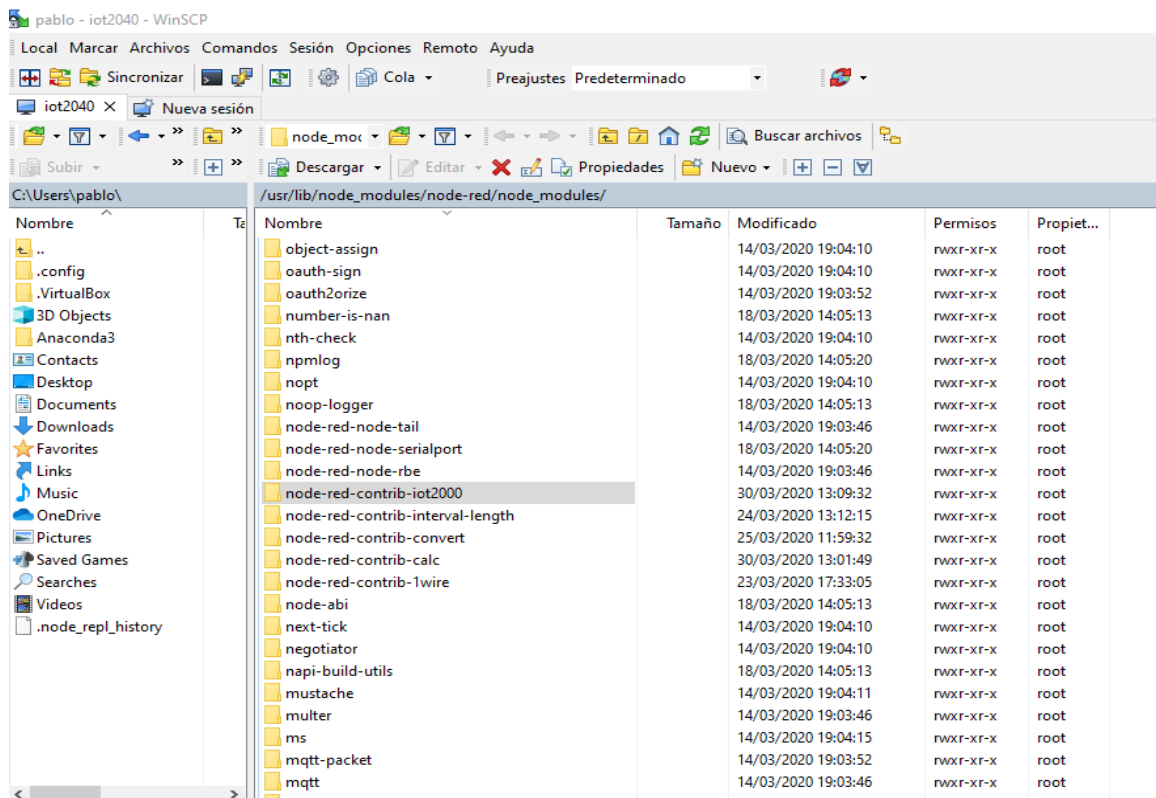
El primer paquete (node-red-contrib-iot2000), para la lectura de las entradas, no se encuentra en el repositorio, por lo que la instalación se tendrá que llevar a cabo de manera manual: copiando directamente el archivo correspondiente en la carpeta de la memoria del IoT2040 donde se encuentran todos los nodos de Node-RED.

Tendremos que recurrir a la herramienta WinSCP, de software libre. Es un cliente SFTP gráfico para Windows que emplea SSH. Con este programa podemos acceder a la plataforma y navegar por las carpetas.

Los nodos instalados se encuentran en la dirección:

`/usr/lib/node_modules/node-red/node_modules/`

Aquí es donde tendremos que añadir la carpeta con los necesarios:



Para la instalación de los otros nodos que necesitamos: calcular intervalos de tiempo y realizar operaciones matemáticas; se puede realizar directamente desde Putty:

- Accediendo primero a la localización de los nodos:
`/usr/lib/node_modules/node-red/node_modules/`
- Instalando con npm:
`npm install node-red-contrib-interval-length`
`npm install node-red-contrib-calc`
`npm install node-red-email`

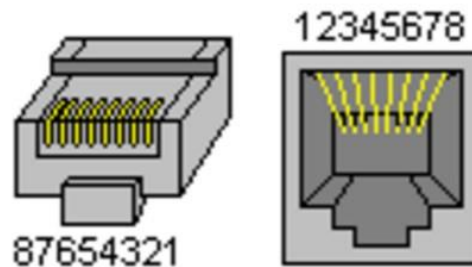
Tras realizar todo este proceso, reiniciamos el dispositivo y después de iniciar el programa nuevamente, ya tendremos añadidos todos los nodos que necesitábamos para construir los flujos.

8.5 ANEXO 5: Conexión y programación de los sensores

En este anexo se expone cómo se ha realizado la conexión de cada uno de los sensores al IoT2040, así como la programación y pruebas experimentales.

Sensor de pulsos:

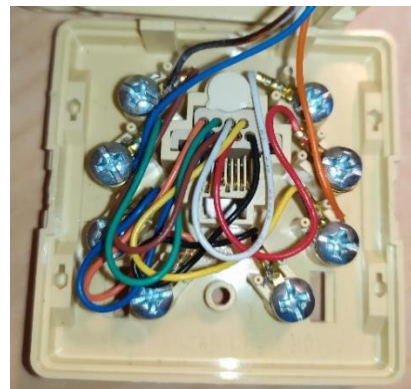
La conexión se realiza a través de RJ45. La distribución de los pines es:



Y los pines que utilizamos son los siguientes:

- Pin 2: Vcc
- Pin 5: GND
- Pin 6: Output

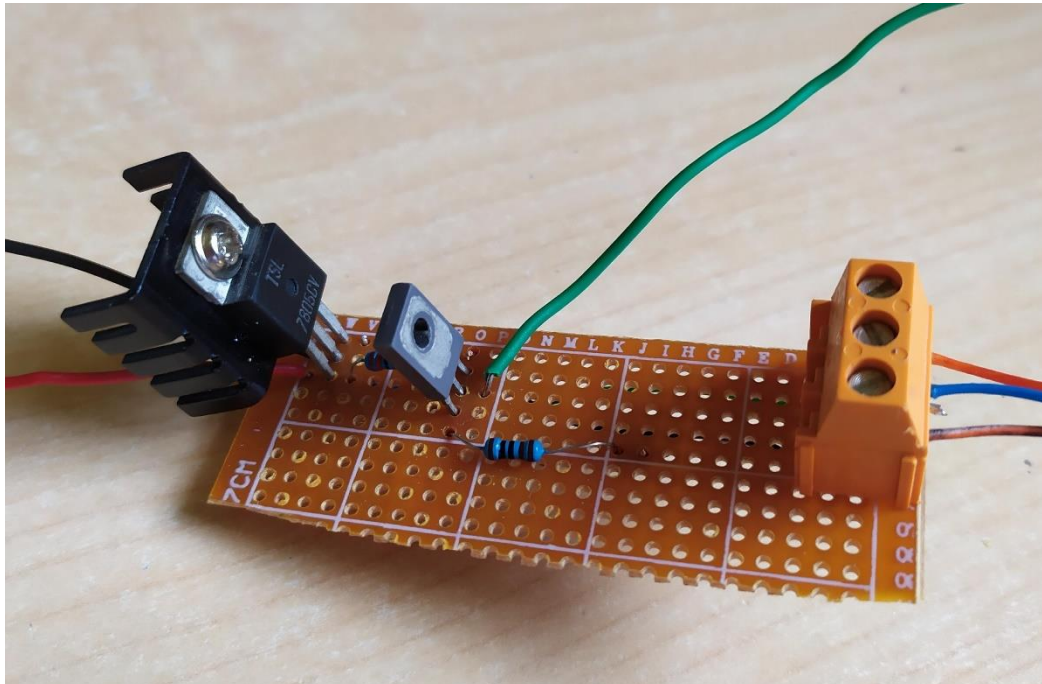
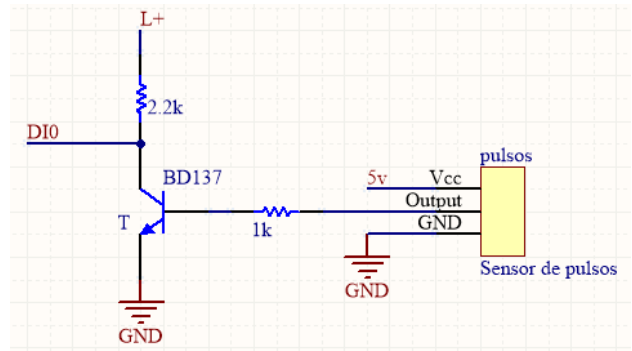
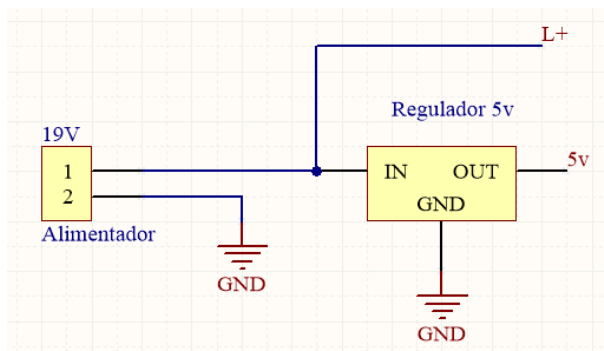
Mediante un conector base hembra RJ45 se separan los cables de los pines. De estos vamos a coger los tres que necesitamos.



Para alimentar el sensor, se coloca un regulador 7805CV, de tensión fija 5V, a partir de la alimentación principal de todo el circuito (los 19V). La señal de salida va a ser como máximo de 5V. El problema que nos encontramos es que, según la documentación de nuestro Simatic

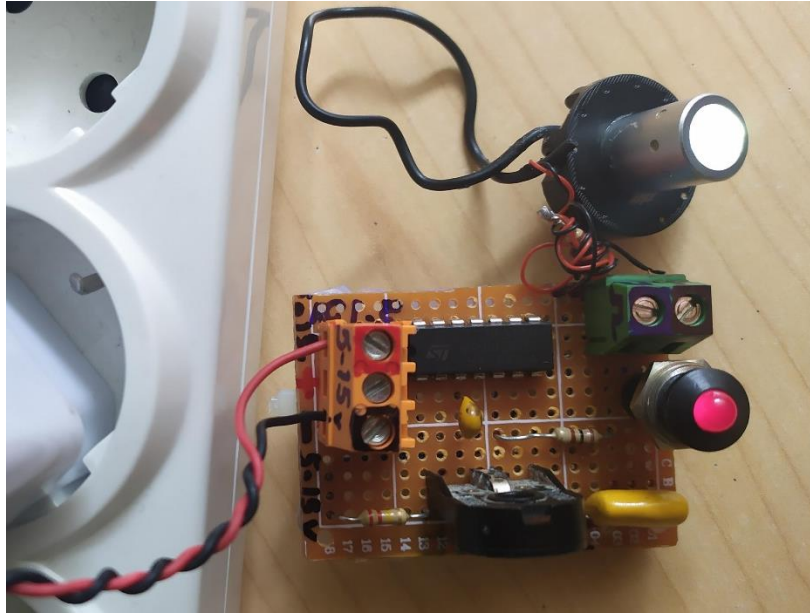
IoT2040, las entradas digitales toman como entrada el valor '1' a partir de 12V. Para resolverlo, vamos a introducir un transistor. Va a estar controlado en la base con la tensión de salida del sensor y según el valor de esta, el transistor estará en corte o en saturación. Conectando el colector a los 19Vdc y el emisor a masa, tendremos una señal de 0 a 19V que podrá reconocer la entrada digital de nuestra plataforma perfectamente.

Los valores calculados y elegidos para los requerimientos de corriente de las resistencias son: resistencia de base = 1k Ω , resistencia de colector = 2'2k Ω . El circuito es el siguiente:



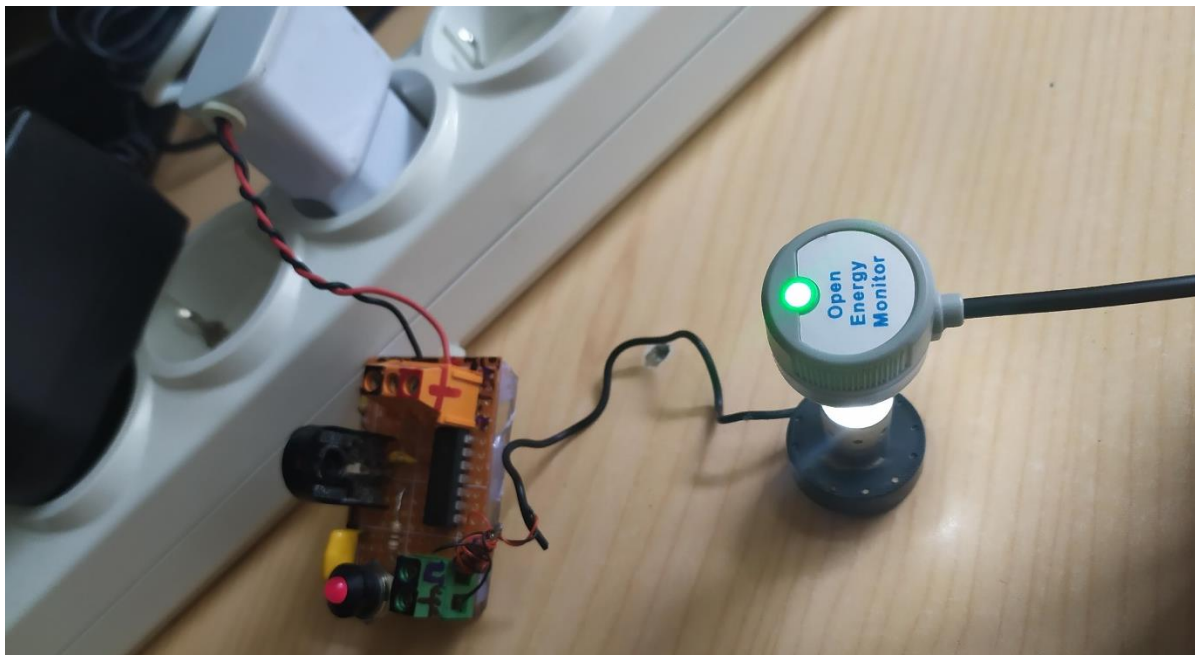
Para simular el funcionamiento de una forma más "real", se ha decidido diseñar y construir un simulador de pulsos de una señal luminosa de frecuencia variable.

Esto se ha hecho con el uso del circuito integrado 555 (realmente un 556, que contiene dos 555 en su interior) como multivibrador astable que genera pulsos cuadrados cuya frecuencia puede ser regulada mediante un potenciómetro. Esto se debe a que la descarga del condensador dependerá de esa resistencia.

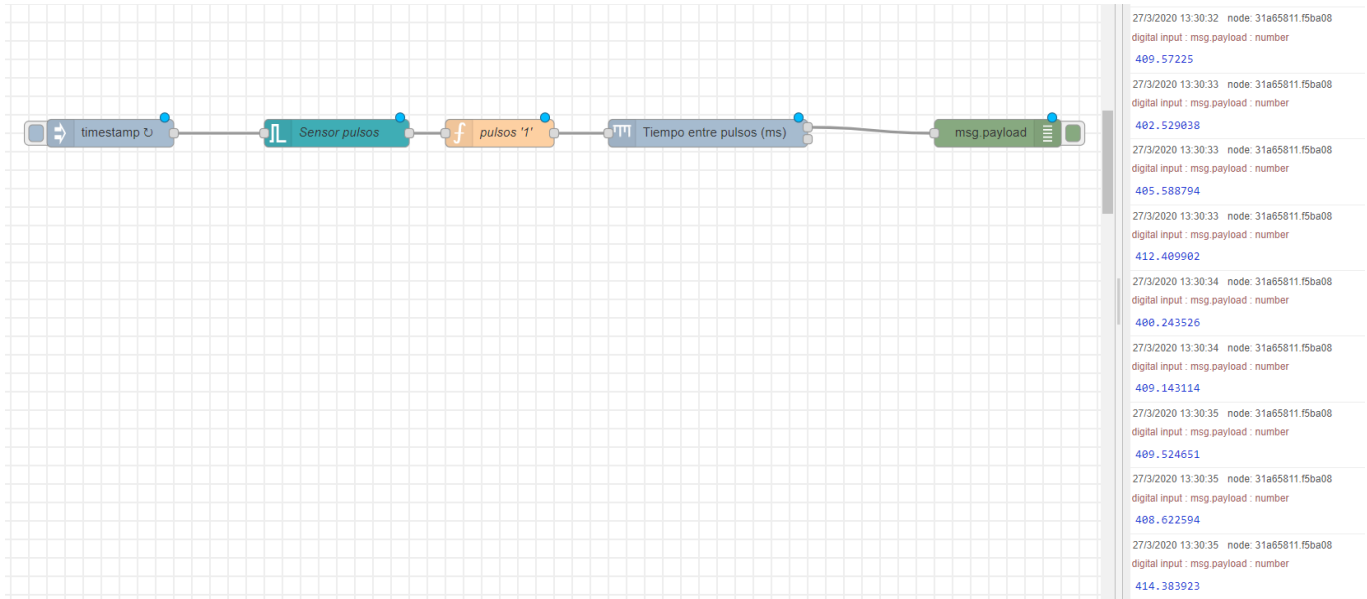


Se conecta en la salida una señal luminosa que podemos usar para que el sensor lea como '1'.

Para la lectura desde el programa se va a conectar la salida del sensor (en este caso el colector del transistor), que tendrá valores '1' o '0', según exista o no pulso, con el pin de entrada digital D10 de nuestro módulo de entradas y salidas.



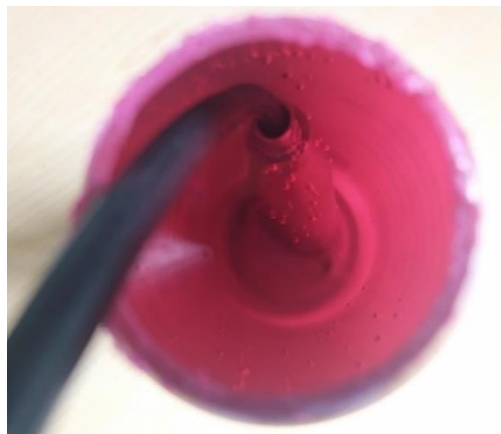
Pasamos a probarlo en Node-RED mediante la lectura constante del puerto digital elegido y comprobando que ante el cambio de iluminación o no (pulso), el programa reacciona correctamente. Sacamos la frecuencia de los pulsos midiendo el tiempo (en milisegundos) entre estos.



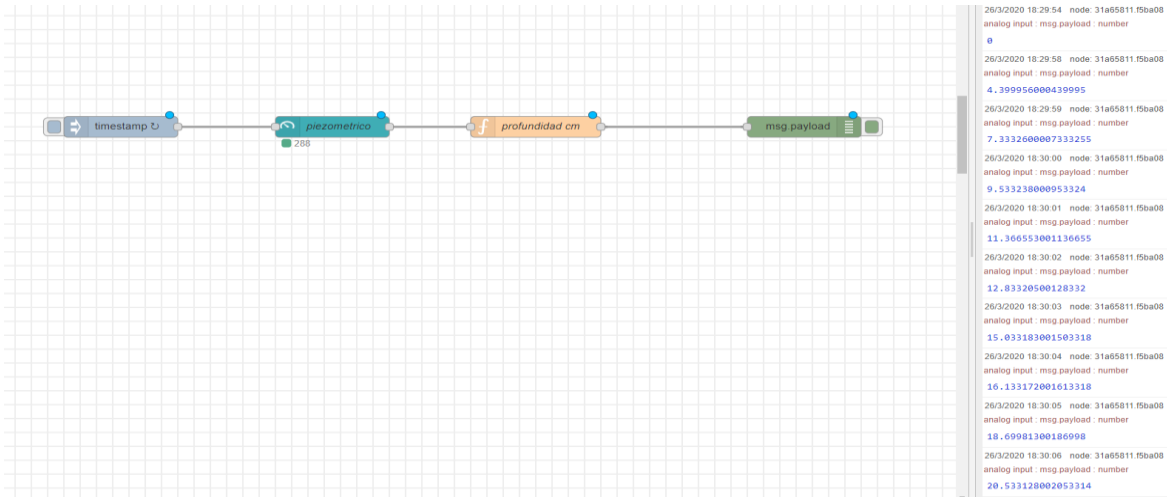
Piezómetro

Dispone de 2 cables: el de alimentación y el que irá a la entrada de corriente analógica I0 de nuestra plataforma.

Para probarlo, introducimos el sensor en un tubo lleno de agua de unos 25 centímetros de profundidad y así se comprueba el aumento proporcional de la corriente al sumergirlo progresivamente, con la ayuda de un polímetro.



Sacamos la ecuación que va a dar la profundidad en función del valor de corriente de la entrada analógica en Node-RED y el programa queda así (en esta captura se muestra el valor de salida en centímetros):



- Nodo profundidad en metros:

```
var piezometrico = msg.payload;

msg.payload=100*(piezometrico-230)/2.7273; //sin multiplicar *100 para pruebas en centímetros

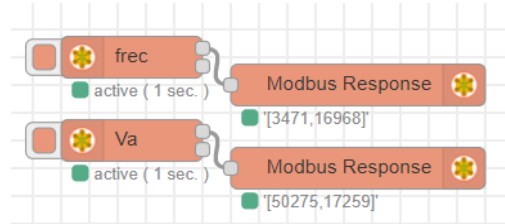
return msg;
```

Analizador de redes

Dispone de 10 pines para poder medir las corrientes y tensiones de las 3 fases a, b y c. Para comunicarse con la plataforma se ha conectado un RS485 al puerto X30 COM (configurándolo previamente vía Putty para establecer que fuese de ese tipo) y se han separado los cables de los pines para poder utilizar los que necesitamos (el 1 y el 2). Se colocará una resistencia de terminación de 120Ω entre los cables de datos A+ y B- para evitar posibles aristas de datos y ruido eléctrico.

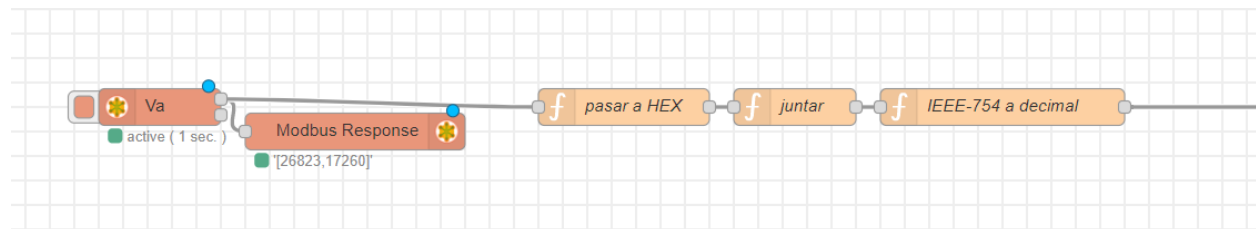


Pasamos ahora al programa de Node-RED para acceder a los valores de los registros de las medidas del analizador. Ajustamos la velocidad a 9600 baudios y configuramos para que acceda con protocolo Modbus. Empezamos leyendo el valor de la tensión de una fase y de la frecuencia y tenemos estos resultados:

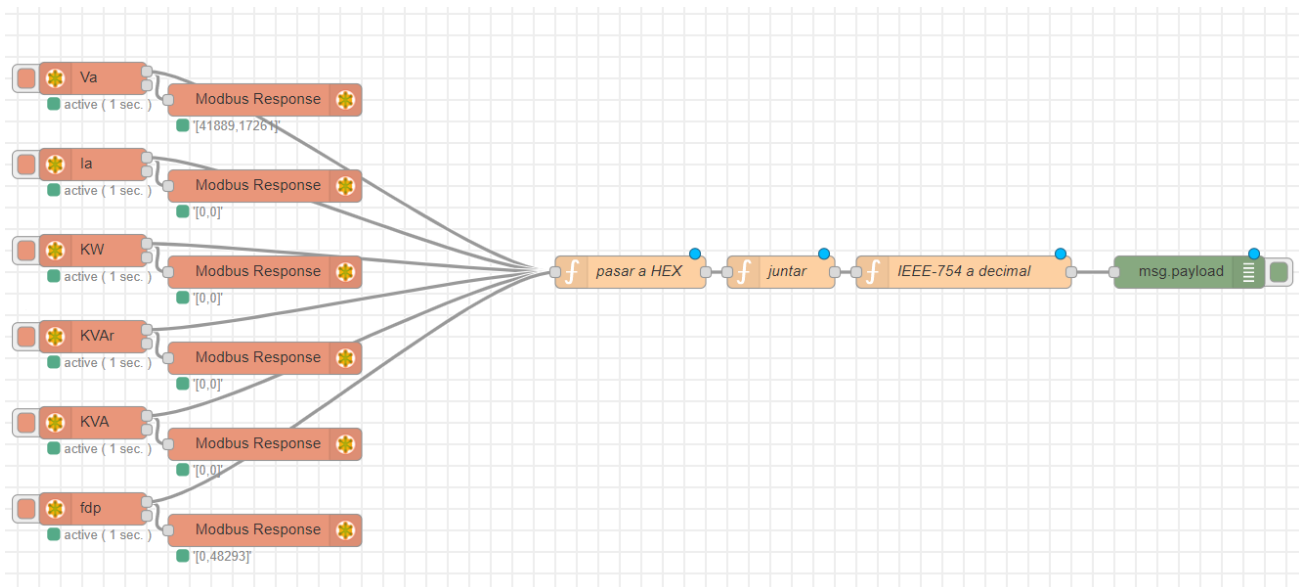


Estos arrays se corresponden a los 2 registros que ocupan ambos parámetros. Los números se corresponden con el valor entero (posición 1) y los decimales (posición 2). Los valores de las lecturas siguen el estándar IEEE-754 coma flotante.

Para convertir estos datos a decimal primero los pasaremos a hexadecimal por separado, los juntaremos, y finalmente convertiremos los 8 dígitos de la norma IEEE-754 a su valor en decimal.



Esto hay que hacerlo para todos los registros que deseamos leer: Tensión, corriente, potencia activa, potencia reactiva, potencia aparente y factor de potencia. El programa en Node-RED quedará de la siguiente forma:



Para simplificar, se agruparán en *subflows* para la obtención de cada parámetro eléctrico.

- **Nodo pasar a hexadecimal y juntar:**

```
msg.payload[0] = msg.payload[0].toString(16);  
msg.payload[1] = msg.payload[1].toString(16);  
msg.payload = '0x' + msg.payload[1] + msg.payload[0];  
return msg;
```

- **Nodo pasar a decimal:**

```
let valor = Number(msg.payload);  
var sign = (valor & 0x80000000) ? -1 : 1;  
var exponent = ((valor >> 23) & 0xFF) - 127;  
var significand = (valor & ~(-1 << 23));  
  
if (exponent == 128)  
    return sign * ((significand) ? Number.NaN :  
    Number.POSITIVE_INFINITY);  
  
if (exponent == -127) {  
    if (significand === 0) return sign * 0.0;  
    exponent = -126;  
    significand /= (1 << 22);  
} else significand = (significand | (1 << 23)) / (1 << 23);  
  
msg.payload = sign * significand * Math.pow(2, exponent);  
  
return msg;
```

- Nodo transformar medida a m.c.a.:

```
msg.payload=10.2*0.06895*(msg.payload/20+0.5)//mca
return msg;
```

Horas de funcionamiento

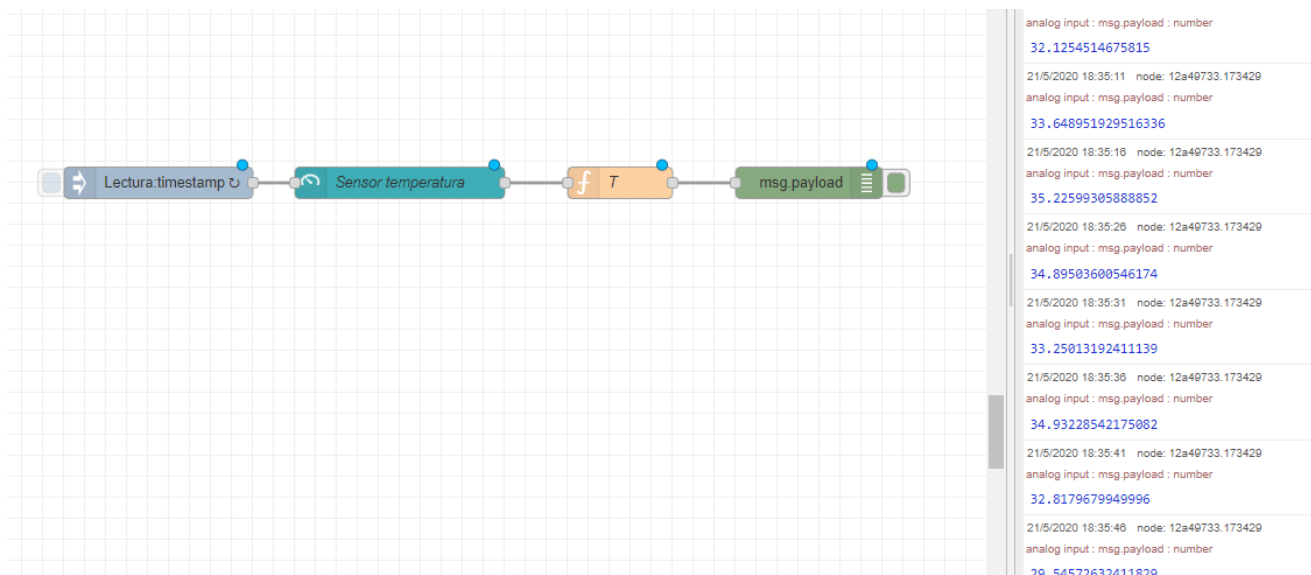
Para contar las horas, se programa un nodo que devuelva el tiempo total transcurrido en horas.

- Nodo horas de funcionamiento:

```
msg.payload = (1/60) + (flow.get('t') || 0);
var a=msg.payload;
flow.set('t', a);
msg.payload=msg.payload.toFixed(2);
return msg;
```

Sensor de temperatura

Se tiene un termistor NTC en un módulo KY-013. Se conectará una patilla a la alimentación de 5v, y la otra a una entrada analógica de corriente del IoT2040. Al cambiar la temperatura, la resistencia del termistor cambiará, por lo tanto, la corriente (a mayor temperatura, menor resistencia y más corriente). Se comprueba con la ayuda de un polímetro.



Subimos la temperatura manualmente y comprobamos su correcto funcionamiento en el programa de Node-RED.

Sensor de presión

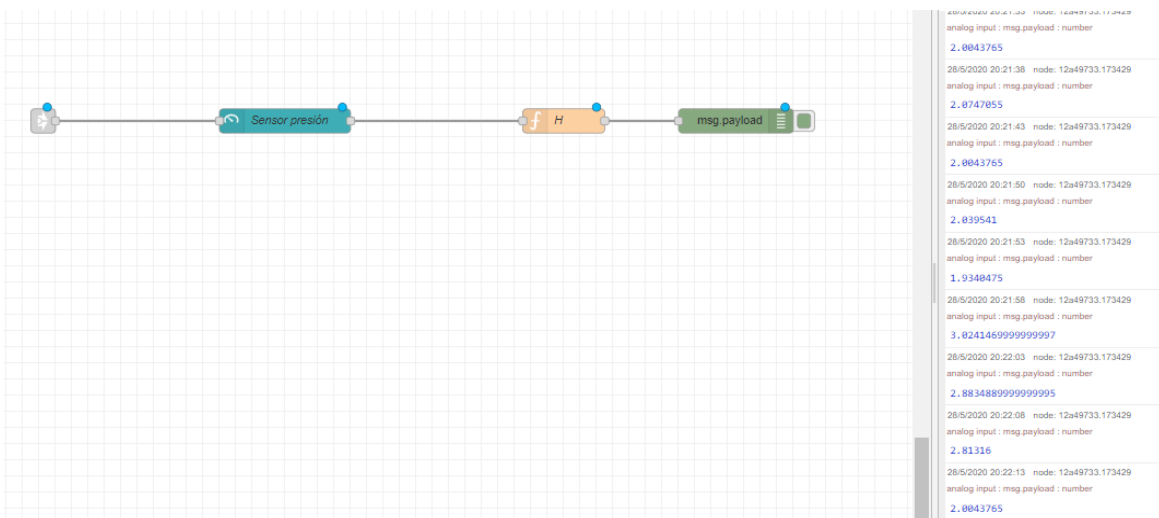
Este sensor dispone de 3 cables: alimentación, masa, y salida de tensión. Se alimenta a 5Vdc, por lo que lo conectaremos a la salida del regulador 7805CV. La salida del sensor se conectará a la entrada U0 de la pasarela inteligente.

Sabiendo que la salida es lineal podemos comprobar las lecturas usando el polímetro. Su rango de medida es de 0-80 psi (0-56.202 m.c.a.) y la salida va de 0'5 a 4'5v.

Para comprobarlo utilizamos un compresor, conectándolo al sensor y viendo cómo responde:



Probando el flujo en Node-RED se comprueba su correcto funcionamiento al variar la potencia del compresor:



8.6 ANEXO 6: Código en JavaScript de los nodos en Node-RED

En este anexo se expone el código en JavaScript que contienen diversos nodos o bloques del programa.

Código de la algoritmia de los KPI's:

- Rendimiento real:

```
pteor=0.736*flow.get('cau')*flow.get('alt')/75;  
msg.payload=100*pteor/flow.get('pot');  
flow.set('rto', msg.payload);  
msg.payload=msg.payload.toFixed(2);  
return msg;
```

- Desviación del rendimiento:

```
msg.payload=100*((flow.get('rto')/flow.get('rtoteor'))-1);  
msg.payload=msg.payload.toFixed(2);  
return msg;
```

- Ratio consumo:

```
msg.payload=flow.get('pot')/(3.6*flow.get('cau'));  
flow.set('ratioc', msg.payload);  
msg.payload=msg.payload.toFixed(2);  
return msg;
```

- Caudal – caudal recomendable:

```
msg.payload=flow.get('cau')-flow.get('caurec');  
msg.payload=msg.payload.toFixed(2);  
return msg;
```

- **Ratio de eficiencia:**

```
msg.payload=1000*flow.get('ratioc')/flow.get('alt');  
msg.payload=msg.payload.toFixed(2);  
return msg;
```

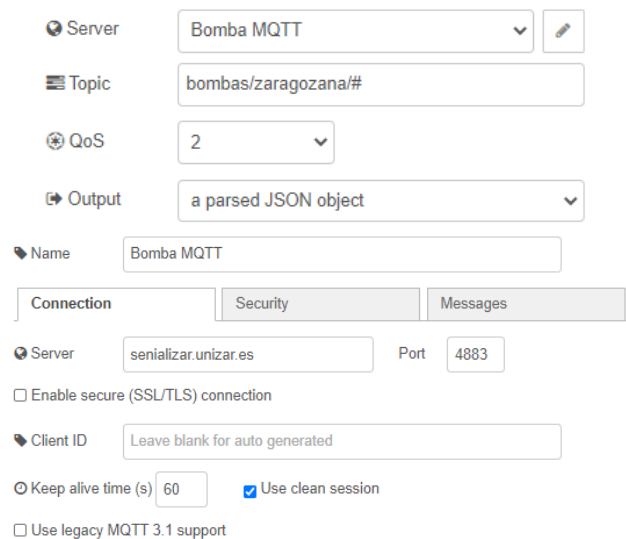
- **Eficiencia hidráulica:**

```
msg.payload=100*flow.get('desteor')/flow.get('des');  
msg.payload=msg.payload.toFixed(2);  
return msg;
```

- **Caudal específico:**

```
msg.payload=flow.get('cau')/flow.get('des');  
msg.payload=msg.payload.toFixed(2);  
return msg;
```

Código de recogida de datos reales vía MQTT:



The image shows a configuration interface for an MQTT client. It includes fields for Server (Bomba MQTT), Topic (bombas/zaragoza/#), QoS (2), and Output (a parsed JSON object). Below these are tabs for Connection, Security, and Messages. The Connection tab is active, showing fields for Server (serializar.unizar.es), Port (4883), Name (Bomba MQTT), Client ID (Leave blank for auto generated), Keep alive time (60), and Use clean session (checked). There is also a checkbox for Use legacy MQTT 3.1 support.

- **Descenso:**

```
if(msg.payload.length===3){ //piezo
    msg.payload=msg.payload[2] ["v"];
    return msg;
}
if(msg.payload!==0){
msg.payload=msg.payload-10.6; //Nivel estatico
flow.set('des',msg.payload);
msg.payload=msg.payload.toFixed(2);
return msg;
}
```

- **Potencia**

```
if(msg.payload.length===2){ //potencia
    msg.payload=msg.payload[0] ["v"];
    return msg;
}
if(msg.payload!==0){
msg.payload=1.457*msg.payload;
flow.set('pot',msg.payload);
msg.payload=msg.payload.toFixed(2);
return msg;
}
```

- **Caudal:**

```
if(msg.payload.length===1){ //caudal
```

```

msg.payload=msg.payload[0] ["v"];

return msg;

}

if(msg.payload!==0){

    if(msg.payload<1000){ //Posible dato anomalo

        msg.payload=1000*msg.payload/5/60;//llegan cada 5min

        flow.set('cau',msg.payload);

        msg.payload=msg.payload.toFixed(2);

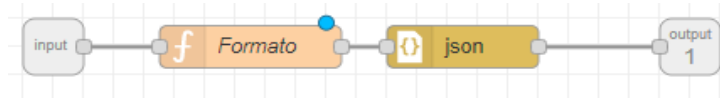
        return msg;

    }

}

```

Código de conversión a formato SenML (ejemplo en caudal específico, para el resto de los datos es igual, pero cambiando nombre, valor y unidades):



```

msg.payload='[{"bn": "urn:dev:org:bomba/", "n": "caudal_especifico",
"u": "l/s/m", "v": ' + msg.payload + '}]';

return msg;

```

Código de aviso por correo electrónico (para ejemplo de caudal específico):

Server	<input type="text" value="smtp.gmail.com"/>
Port	<input type="text" value="465"/> <input checked="" type="checkbox"/> Use secure connection.
Userid	<input type="text" value="pablozgz96@gmail.com"/>
Password	<input type="password" value="....."/>
Use TLS?	<input checked="" type="checkbox"/>

```
if (msg.payload>8){  
  dato=msg.payload;  
  mensaje = "Caudal específico fuera de rango: " + dato + " l/s/m"  
+ "\n";  
  msg={  
    payload: mensaje,  
    to: "pablozgz96@gmail.com",  
  };  
  return msg;  
}
```