

Uso de tecnologías web para la entrega de datos de sensado de una red de distribución de agua usando la placa de desarrollo Intel Galileo

Juan Antonio Alvarado Elías, Roberto Solís, Oscar Osvaldo Ordaz García,
José Guadalupe Arceo Olague

Universidad Autónoma de Zacatecas,
Zacatecas, México

tonnyalvarado94@gmail.com, rsolis@uaz.edu.mx, oscarord27@hotmail.com,
arceojg@uaz.edu.mx

Resumen. Este trabajo expone el análisis, diseño e implementación de un sistema de procesamiento y presentación de datos de presión, flujo y nivel de agua de una Red de Distribución de Agua Potable (RDAP) para su uso en el Internet de las Cosas a través de una placa de desarrollo Intel Galileo. Se establecen los componentes principales del procesamiento del sistema inteligente y los elementos utilizados por la placa de desarrollo Intel Galileo. El diseño y desarrollo del procesamiento de datos se evaluó realizando pruebas de adquisición de voltaje con una fuente variable y se verificaron los resultados experimentales obtenidos. La presentación de datos se compara con los valores esperados en respuesta a la entrada de voltaje en contraste con los datos que se visualizan en un dispositivo LCD (Liquid Crystal Display) y en páginas web apoyándose en tecnologías web como Dweet.io, Freeboard.io y Telegram.

Palabras clave: internet de las cosas, placa de desarrollo Intel Galileo, tecnologías web, dweet.io, freeboard.io, telegram.

Use of Web Technologies for the Delivery of Sensing Data in a Drinking Water Distribution Network Using the Intel Galileo Development Board

Abstract. This work exposes the analysis, design, and implementation of a system that processes and presents the pressure, flow and water level data of a Drinking Water Distribution Network (DWDN) for its use in the Internet of Things through an Intel Galileo development board. The main components for the intelligent system processing and the elements used by the Intel Galileo development board are established. The design and development of data processing was evaluated by performing voltage acquisition tests with a variable voltage source and the experimental results obtained were verified. The presentation of data is compared with the expected values in response to the voltage input in contrast to the

data displayed on an LCD (Liquid Crystal Display) device and on web pages based on web technologies such as Dweet.io, Freeboard.io and Telegram.

Keywords: Internet of Things, Intel Galileo Development Board, Web Technologies, Dweet.io, Freeboard.io, Telegram.

1. Introducción

El sistema de distribución de agua potable es deficiente en las ciudades como Zacatecas, colonias se quedan sin agua diariamente y existen fugas en diferentes espacios en donde se pierde aproximadamente el 40 por ciento del agua potable [1]. Estos problemas podrían minimizarse si existe una forma de obtener mediciones en tiempo real de parámetros como presión, flujo y nivel de agua para la toma de decisiones ya que permiten localizar posibles fugas mediante algoritmos de detección [2].

Actualmente se han realizado varios trabajos para obtener mediciones de los parámetros citados mediante dispositivos lógicos programables [3], sin embargo, es requerido mejorar la sensibilidad, estabilidad y la presentación de los datos para poder implementarse en una RDAP.

Este trabajo está enfocado en procesar y presentar los resultados obtenidos de las lecturas adquiridas en cada ADC (Analog to Digital Converter) de una placa de desarrollo Intel Galileo [4]. Los voltajes hacen referencia a los parámetros que se miden con sensores de presión, flujo y nivel de agua. Para la interpretación de los datos se realiza su procesamiento en Python y después se despliegan los resultados mediante el módulo LCD y mediante las tecnologías web Dweet.io [5], Freeboard.io [6] y Telegram [7]. Dweet.io es un sistema de mensajes y alertas para el Internet de las Cosas, Freeboard.6 es una herramienta de visualización simple y fácil de usar, Telegram es una app de mensajes para desktops y móviles basada en la nube.

El documento está organizado de la siguiente manera: primero se comentan los antecedentes y los trabajos previos; posteriormente se explican los materiales y los métodos para la descripción de la arquitectura, procesamiento y representación de datos; enseguida se comentan los resultados y finalmente las conclusiones.

2. Antecedentes y trabajos previos

Actualmente existen instrumentos que permiten realizar lecturas de diferentes parámetros en una RDAP y su comportamiento. El manómetro mide la presión, el rotámetro mide el flujo y el visor de nivel hidrostático mide el nivel de agua [8, 9, 10]. Así como existen instrumentos para las diferentes mediciones que se realizan en una RDAP, también existen sensores que realizan la conversión de los parámetros físicos en corriente eléctrica o voltaje, que deben ser caracterizados. La recolección manual de estos parámetros es ineficiente para la toma de decisiones en la distribución del agua, dado que la precisión y el tiempo requerido para tomar las muestras dependen de la pericia del capturista [11], por lo que monitorear las magnitudes de los sensores a través de una

placa de desarrollo Intel Galileo supone una ventaja para la toma de decisiones en la distribución de agua.

Se han realizado trabajos en Arduino [12, 13] y FPGA [14] para resolver algunos problemas de adquisición y presentación de datos referentes a una RDAP; sin embargo, no los han resuelto del todo, ya que se han encontrado diferentes problemas de conversión, procesamiento y presentación. Este trabajo resuelve los problemas de procesamiento y presentación, mediante el uso de la placa de desarrollo Intel Galileo y el uso de tecnologías Web. Los aspectos a resolver son el procesamiento del cálculo matemático para cada lectura asegurando la asertividad de los valores y la generación de la presentación de resultados en tiempo real mediante tecnologías Web que permita el fácil acceso a la información.

El sistema para la adquisición de datos de presión, flujo y nivel de agua, se compone de cuatro partes fundamentales: (1) la lectura de las magnitudes obtenidas por los sensores, (2) el acoplamiento de los datos obtenidos de los sensores a la placa de desarrollo Intel Galileo mediante un circuito conversor Corriente (I) a Voltaje (V), (3) el procesamiento de los datos obtenidos de los sensores ahora convertidos en magnitud de voltaje, y (4) la presentación de datos mediante el dispositivo LCD, la página de visualización de datos Freeboard.io con ayuda del servicio de mensajes de Dweet.io y la aplicación Telegram. Las cuatro etapas se pueden ver en la Fig. 1.

En este trabajo se presentan las secciones de procesamiento y presentación de datos obtenidos mediante la placa de desarrollo Intel Galileo, para las cuales se necesitan los materiales descritos en la siguiente sección.

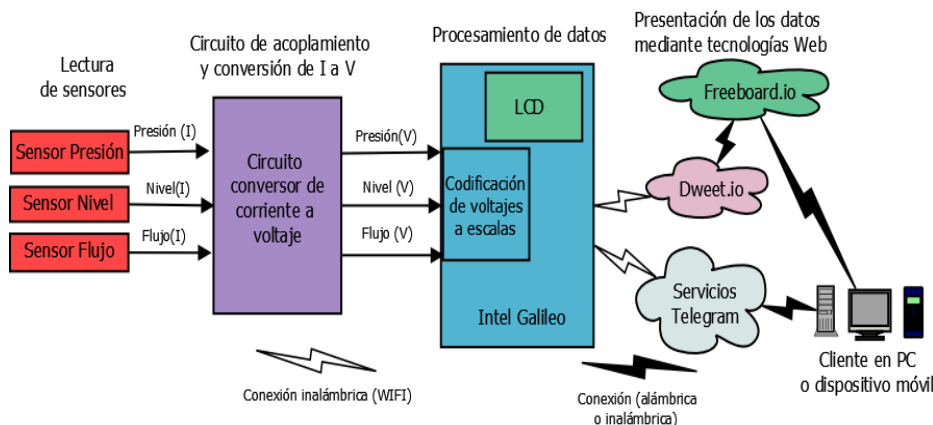


Fig. 1. Diagrama de bloques del sistema de obtención de datos.

3. Materiales y métodos

En el proyecto se utilizó la placa de desarrollo Intel Galileo, un Grove Starter Kit compatible con la placa de desarrollo, una unidad extraíble la cual es una memoria microSD desde la cual arranca el sistema operativo instalado, el cual es Linux en la distribución llamada Yocto, utilizada para Internet de las Cosas, un cable On-the-go

(OTG) y un dispositivo WIFI USB para conexión inalámbrica. La placa de desarrollo Intel Galileo contiene un procesador Intel Quark SoC X1000 de 32 bits que funciona a una frecuencia de 400 MHz, la placa dispone de un puerto USB host, un puerto USB cliente, un puerto RS-232 y un zócalo para tarjetas microSD con capacidad de soportar memorias de hasta 32 GB.

La placa dispone de un voltaje de operación configurable entre 3.3 V y 5 V, además de 6 puertos de entrada analógica con un circuito conversor analógico/digital (ADC) modelo AD7298 de Analog Devices con una resolución configurable de 10 bits o 12 bits. La placa tiene la capacidad de soportar un sistema operativo Linux instalado desde una memoria microSD.

El Grove Starter Kit contiene un Shield o placa de acoplamiento con voltaje de operación de 3.3 V y 5 V; conjuntamente contiene un dispositivo LCD [15]. Las especificaciones de la placa determinan que la memoria microSD que soporta es de tipo SD/HC, de clase 4 con capacidad máxima de 32 GB y mínima de 8 GB, la cual debe ser configurada para iniciar desde un dispositivo USB con el sistema operativo Linux especial para Galileo [16].

El cable OTG se recomienda con especificación Hub 2.0, pero puede soportar la especificación Hub 1.0, esto cuando sea compatible con el dispositivo WIFI USB. Para la conexión inalámbrica, es necesario un dispositivo WIFI USB con velocidad de 150Mbps o 300Mbps, con protocolo de conexión 802.11n, siendo necesario para casos especiales instalar los controladores adicionales dependiendo del modelo de dispositivo WIFI USB.

Para la programación del sistema se utilizó una computadora con el controlador FT231 USB UART instalado, para el reconocimiento de la placa de desarrollo Intel Galileo, y el software de conexión serial configurable a una velocidad de 115200 bits por segundo. Es necesario tener una conexión a Internet para la interacción con el sistema de mensajes Dweet.io así como con los servicios de Telegram.

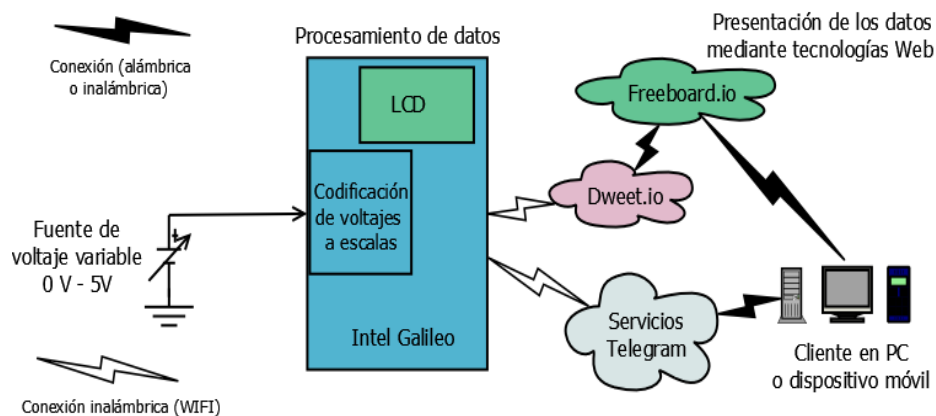


Fig. 2. Diagrama general del proyecto.

El proyecto se realizó utilizando el lenguaje de programación Python [17], el cual es un lenguaje de código abierto, portable, orientado a objetos y dinámico, ya que tiene la

posibilidad de soportar módulos de diferentes lenguajes. Además, posee librerías y características poderosas adaptables a las diferentes necesidades de conexión a través de Internet.

Para conectar todos los materiales anteriores primero debe realizarse la correcta instalación del sistema operativo Linux en una tarjeta micro SD. Para instalarse debe insertarse en la ranura de tarjetas de la placa de desarrollo Intel Galileo y conectarse a la alimentación de voltaje. Acto seguido se debe conectar a la computadora e instalar las librerías de Python requeridas.

Los directorios se manejan igual que en un sistema operativo Linux, solamente debe crearse un directorio de trabajo y de manera opcional se puede trabajar con la herramienta *git* en cualquier plataforma que preste este servicio.

El diagrama general del proyecto se muestra en la Fig. 2, que se enfoca en las etapas de procesamiento y presentación de datos.

Para el análisis de la etapa de procesamiento, se consideraron tres sensores con rango de salida de corriente continua (CC) de 4 a 20 mA:

- Sensor Serie 626 Pressure Transmitter de Dwyer para medir presión con un rango de 0 a 40 Bar de presión, con una precisión de 0.25%. Requiere alimentación de 10 a 30 V_{cc} [18].
- Sensor 2551 de Signet para medir el flujo con un rango de 0.5 a 10 m/s con una precisión de $\pm 1.0\%$. Requiere alimentación de 24 V_{cc} ($\pm 10\%$) [19].
- Sensor ultrasónico serie ToughSonic® modelo TSPC-30S1-232 de SENIX para medir la altura libre de agua en tanques con un rango de 10.16 a 426.72 cm, con precisión del 0.5%. Requiere alimentación de 15 a 30 V_{cc} [20].

En la Fig. 3 se muestra gráficamente cómo se desarrolla el flujo de información considerado para el funcionamiento de las etapas de procesamiento y presentación. Inicialmente se realiza la lectura de cada uno de los ADC, para que tales lecturas sean procesadas por el primer módulo de programa y sean presentadas en el LCD.

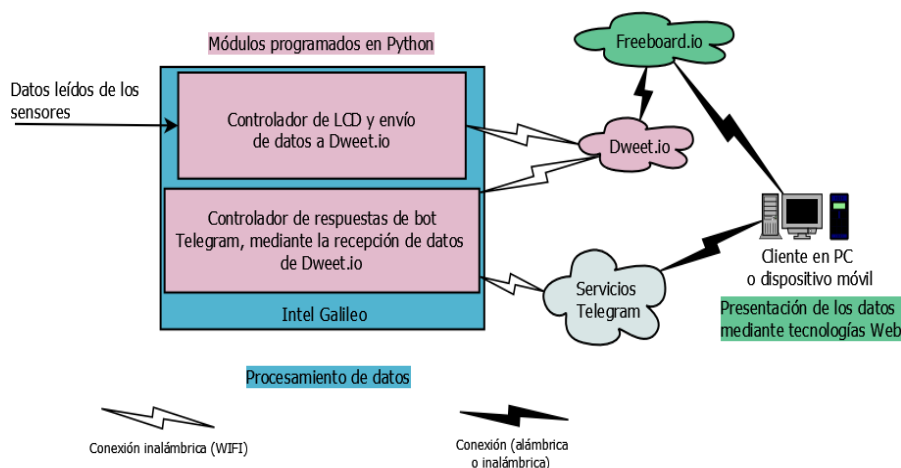


Fig. 3. Flujo de información dentro del micro controlador Galileo.

Una vez realizado el procesamiento de los datos, éstos son enviados a Dweet.io. El nombre Dweet nace de una analogía a Twitter en donde el principio es publicar datos desde “cosas” o aparatos electrónicos conectados a Internet [5]. El segundo módulo del programa, se encarga de la atención exclusiva a un bot de Telegram [21]. El bot es un módulo programado que permite realizar respuestas predefinidas para una aplicación específica. Telegram cuenta con bots en su servicio. Estos son pequeños programas que se encuentran en los servidores y pueden ser “llamados” en cualquier momento para conseguir cientos de funcionalidades diferentes para ser usadas en los chats en la aplicación. Además de estos módulos, se genera otro programa para obtener la lectura de los puertos analógicos A0, A1 y A2, que es donde se conectan los circuitos que convierten la I a V, proveniente de las señales de los sensores de presión, flujo y nivel respectivamente. Para obtener una mejor precisión, cada ADC se configura con una resolución de 12 bits, por lo que se pueden obtener 4,096 valores diferentes de voltaje. Después de tomar la lectura, se realiza el cálculo correspondiente a cada ADC y se almacena en una variable que identifica cada parámetro, para ser presentada en el dispositivo LCD y se agrupan los valores para enviarse a Dweet.io.

Tabla 1. Valores mínimos y máximos de medición de los sensores.

Sensor	Rango del Sensor
Presión	0 – 40 bares
Flujo	0.5 – 10 m/s
Nivel	10.16 – 426.72 cm

En la Tabla 1, se muestran los valores de los parámetros mínimo y máximo que son capaces de detectar los diferentes sensores, además de las unidades de medición, los cuales se especifican en las fichas técnicas correspondientes. El rango de voltaje para las entradas de cada ADC fue configurado de 0 a 5 V. Con estas consideraciones se realiza el cálculo matemático con la relación proporcional directa en ecuación **¡Error! No se encuentra el origen de la referencia.:**

$$LS = \left(\frac{V_e}{V_{max}} * (S_f - S_i) \right) + S_i, \tag{1}$$

dónde:

- LS: Lectura del sensor
- Ve: Voltaje de entrada a convertir,
- Vmax: Voltaje de salida máximo,
- Sf: Valor máximo de escala del sensor,
- Si: Valor mínimo de escala del sensor.

Considerando la relación matemática obtenida, se realiza el diseño de un algoritmo para encargarse de la lectura de los valores enviados por los sensores a cada ADC y entonces realizar el procesamiento de datos. Para el código del bot de Telegram se utiliza una plantilla Open Source, realizando las adecuaciones necesarias para cumplir con las expectativas del proyecto.

Para la presentación de datos usando la herramienta web Freeboard.io es necesario realizar el registro de una cuenta en su sitio web. Con la cuenta asignada se puede ingresar con el usuario y contraseña respectivos. Freeboard.io despliega de manera visual

la información enviada a Dweet.io, por lo cual es de vital importancia para el diseño de sistema, entender cómo funciona tal servicio y como debe estructurarse e identificarse la información que se le envía. La Fig. 4 muestra un ejemplo de cómo se despliegan los datos en Freeboard.io.

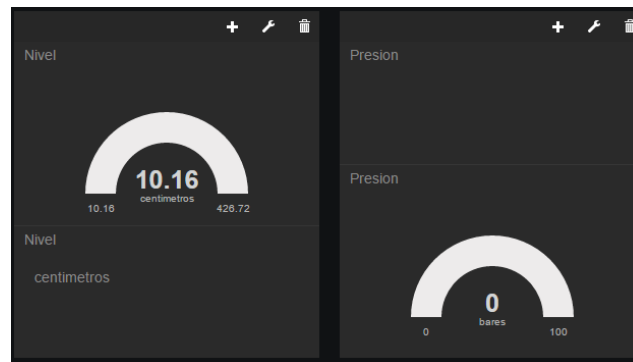


Fig. 4. Ejemplos de paneles en la herramienta web Freeboard.io.

Después de realizar la codificación de los módulos programables, el módulo dedicado al bot de Telegram se ejecutó en modo “background”, de esta manera el módulo encargado de la lectura de cada ADC se puede ejecutar independientemente del módulo dedicado al bot de Telegram. Así se asegura que el sistema se ejecute de manera completa y correcta.

4. Resultados

Las herramientas utilizadas para Internet de las Cosas ofrecen una variedad de formas de trabajar, por lo que es aconsejable utilizar el ambiente que se sugiera para cada una de ellas [22]. Gracias a estas herramientas, los medios de procesamiento y presentación tienen una confiabilidad evidente, los cuales pueden ser consultados en la Tabla 2, donde se demuestra la precisión obtenida en los diferentes medios de presentación.

La ejecución del sistema después de ser analizada se estableció en 3 segundos por iteración del ciclo del algoritmo, primero se realiza el envío de datos y después se realiza una pausa al sistema, la razón de esto es, que el sistema tenga un tiempo para enviar la respuesta. Con 2 segundos se generan retrasos en la ejecución y un entorpecimiento en la entrega de datos desde las aplicaciones web. Por este inconveniente se estableció un tiempo mayor. Sin embargo, con 4 segundos el sistema se torna lento considerando que se espera que la información sea la más reciente. Así fue que probando con iteraciones de 3 segundos se comprobó que es un tiempo ideal con retrasos mínimos y que permite continuar con fluidez el sistema.

La conexión al bot de Telegram tiene como gran ventaja la disponibilidad de información, ya que es una aplicación muy versátil. Pensar en Telegram es considerar que se puede consultar información importante para los interesados del sistema en cualquier

momento y en cualquier lugar. En la Fig. 5 se muestran algunos de los valores solicitados con un teléfono inteligente que tiene instalada la aplicación Telegram; en ella también se puede observar la forma de solicitar cada dato según sea necesario.

Tabla 2. Resultados para cada medio de presentación.

V _e	Valor ideal del sensor			Valor en LCD			Valor en Freeboard			Valor en Telegram		
	P	F	N	P	F	N	P	F	N	P	F	N
0	0	0.5	10.16	0	0.5	10.16	0	0.51	10.16	0	0.51	10.16
0.5	4	1.45	51.81	4.01	1.35	51.8	4.02	1.35	51.81	4.02	1.35	51.81
1	8	2.4	93.47	8.02	2.5	93.57	8.01	2.5	93.57	8.01	2.5	93.57
1.5	12	3.35	135.12	12.01	3.3	135.2	12.02	3.29	135.19	12.02	3.29	135.19
2	16	4.3	176.78	16.03	4.04	176.69	16.02	4.05	176.68	16.02	4.05	176.68
2.5	20	5.25	218.44	40.05	5.24	218.55	40.04	5.25	218.55	40.04	5.25	218.55
3	24	6.2	260.09	24.05	6.04	260.24	24.04	6.05	260.25	24.04	6.05	260.25
3.5	28	7.15	301.75	28.05	7.04	301.54	28.05	7.05	301.55	28.05	7.05	301.55
4	32	8.1	343.4	32.04	8.03	343.5	32.05	8.02	343.51	32.05	8.02	343.51
4.5	36	9.05	385.06	36.07	9.03	385.2	36.06	9.02	385.21	36.06	9.02	385.21
5	40	10	426.72	39.99	9.99	426.61	39.99	9.99	426.62	39.99	9.99	426.62

V: Voltaje de entrada P: Presión medida en bares F: Flujo medido en m/s N: Nivel medido en cm



Fig. 5. Solicitud de datos mediante *Telegram*.

El área de trabajo de los sensores se simula con una fuente de poder variable con la que se realizaron pruebas exhaustivas para el procesamiento y presentación de los datos. La conectividad física se muestra en la Fig. 6.



Fig. 6. Pruebas de procesamiento y presentación de datos respecto a cada elemento.

5. Conclusiones

En este trabajo se realiza el procesamiento y la representación de datos referidos a los datos de presión, flujo y nivel de agua. El procesar la información y la presentación de datos en la plataforma de desarrollo Intel Galileo y la codificación con el lenguaje de programación Python, permiten la movilidad y evitan la necesidad de estar frente al dispositivo físico para tomar lecturas del comportamiento de los sensores. La realización de este sistema, permite procesar y representar datos lo cual es el objetivo principal de este trabajo.

Los resultados obtenidos demuestran mejoras de precisión y exactitud al igual que en disponibilidad de los datos comparados con versiones anteriores de este mismo sistema. Uno de los problemas principales a resolver en versiones anteriores del sistema era la latencia con respecto al envío y recepción de datos en las plataformas web. Estos problemas se han resuelto reestructurando el código del programa.

La variación de los valores obtenidos con relación a los valores ideales es mínima y en algunos casos se puede considerar despreciable, esto debido al tipo de dato que se utiliza en la codificación, el cual es de tipo *float*. Cabe recordar que estos tipos de datos son aproximaciones de los valores reales por lo que la variación es insignificante. También se solucionaron ciertos problemas de conexión ya que anteriormente había momentos en los que la conexión fallaba y retrasaba el sistema, sin ser culpa de la conexión a internet precisamente. Esto se debía a ciertas excepciones que no eran controladas

adecuadamente. Al tratar con estas excepciones, las únicas fallas de conexión ahora son provocadas exclusivamente por la falla de la conexión a internet.

Como trabajo futuro se pretende realizar las etapas del sistema de adquisición de datos de presión, flujo y nivel de agua en entornos relevante para validar el prototipo, para lo cual es necesario considerar la tolerancia de error de cada dispositivo y realizar las pruebas para establecer los porcentajes de error del sistema.

6. Recomendaciones

Al continuar con el desarrollo del sistema total, es necesario abordar temas de seguridad, control de accesos y cuentas de usuarios. Para la implementación y puesta en marcha del sistema en una RDAP, es necesario considerar personal con conocimientos básicos sobre sistemas operativos Linux para la administración y acceso al mismo.

En caso de un cambio de modelo de sensores, debe planearse el reemplazo que cumpla con la característica de envío a la salida de 4 a 20 mA, ya que de ser así facilita la reconfiguración del código a la nueva escala de los nuevos sensores.

Referencias

1. Tetreault, V.: El abastecimiento del agua en Zacatecas y Guadalupe ¿Cuál es el problema? en La Jornada Zacatecas. Consultada en <http://ljz.mx/2014/02/26/el-abastecimiento-del-agua-en-zacatecas-y-guadalupe-cual-es-el-problema/> (2014)
2. Informes denominados Actualización de la disponibilidad media anual de agua subterránea, disponibles en la página de Internet de la CONAGUA (<http://www.cna.gob.mx/>)
3. Solís, R. et al.: Beneficios y Desafíos en el uso de una Red Inalámbrica de Sensores para el Monitoreo de una Red de Distribución de Agua Potable, Memorias del CONCYE 2011, 21–23 (2011)
4. Intel. Placa de desarrollo Intel® Galileo Gen 2. [online]. Disponible en: <http://www.intel.es/content/www/es/es/do-it-yourself/galileo-maker-quark-board.html> (2016)
5. Dweet.io: Simple messaging (and alerts) for the Internet of Things. <https://dweet.io/>
6. Freeboard.io: Visualiza el Internet de las Cosas. <https://ricveal.com/blog/freeboard-io/>
7. Telegram: A New Era of Messaging. <https://telegram.org/>
8. Enciclopedia Universal Ilustrada, Europeo-Americana. Tomos XXXII, VII, Hijos de J Espasa editores, Barcelona, España
9. Enciclopedia de la Ciencia y de la Tecnología. Tomo VIII, Editorial Salvat. 1° Edición, Barcelona, España (1964)
10. P. Gerhart, R. Gross, Hochtein J.: Fundamentos de Mecánica de Fluidos. 2° Edición, Addison-Wesley Iberoamericana, USA (1995)
11. Toloza, J., Leiva, L., Riba, A., Carmona, F., Acosta, N.: Desarrollo de un prototipo de sistema portátil para la detección en tiempo-real de la necesidad de riego en cultivos de producción intensiva. En: XVI Congreso Argentino de Ciencias en Computación, 962–971 (2011)
12. Rodríguez, J., Román, J.: Sistema de adquisición de variables hidráulicas en una red de distribución de agua potable utilizando tecnología Zigbee. Universidad Autónoma de Zacatecas (2014)

13. Flow Rate Sensor Interfacing: How to Measure Liquid with an Arduino. <https://diyhack-ing.com/arduino-flow-rate-sensor/>
14. García-Duran, A., Ordaz-García, O., Arce-Olague, J., Hernández-Calviño, M.: Adquisición de datos de sensores de una red de distribución de agua potable. En: Congreso Internacional de Investigación, Academia Journals Córdoba, Veracruz, México, 7(3), 255–259 (2015)
15. Desarrollo de aplicaciones en placas Intel Galileo, Notas del curso impartido por Solís, R., y Ordaz-García, O. Escuela de Verano, Universidad Autónoma de Zacatecas (2016)
16. Galileo Datasheet, Intel 329681-003US (2013)
17. Introducción al Lenguaje Python, Notas del curso impartido por Solís, R. y Ordaz-García, O. Escuela de Verano, Universidad Autónoma de Zacatecas (2016)
18. Series 626 & 628 Pressure Transmitters. Specifications - Installation and Operating Instructions. Bulletin E-111, Dwyer Instruments, Inc. Michigan City, Indiana 46361 USA (2010)
19. Magmeter 2551 Manual. Signet- Senninger LLC, Medidor ciego de flujo electromagnético 2551. 3-2551.090 Rev. L 10/12 (2012)
20. ToughSonic® TSPC Series. Ultrasonic Distance Sensors. Push-button Teachable and/or PC Configurable, Senix Corporation., Rev. 6 (2011)
21. Llegan los bots a Telegram: cómo crear el tuyo propio. <https://www.xatakamovil.com/aplicaciones/llegan-los-bots-a-telegram-como-crear-el-tuyo-propio>
22. Arce, A.: Internet of things. Curso de capacitación de LABSOL (2016)