

DISTRIBUCIÓN DE PROCESOS EN REDES INTELIGENTES PARA EL INTERNET DE LAS COSAS

DISTRIBUTION OF PROCESSES IN INTELLIGENT NETWORKS FOR THE INTERNET OF THE THINGS

R. Alvarado ¹

Resumen: El internet de las cosas busca conectar personas, equipos, dispositivos y todo lo imaginable a través de internet, para ello se venían utilizando sencillos dispositivos como sensores, actuadores y microcontroladores. Para que el actuador responda de una manera lógica a una medida tomada por un sensor se utilizan los microcontroladores, los cuales son implementados en placas electrónicas como Arduino, Edinson y Raspberry.

Actualmente existen otras tecnologías y empresas de redes, que realizan la labor del control de los procesos que hace el microcontrolador desde dispositivos de red como los router o Gateway. Mientras las empresas de circuitos embebidos buscan la miniaturización de sus tarjetas y formas para hacer más fácil su programación, empresas de equipos de redes como CISCO, entran al ruedo del internet de las cosas con equipos de red que tienen aplicaciones programables de software a través de nuevos sistemas operativos.

Para realizar un proyecto del internet de las cosas se puede utilizar un modelo de diseño con sistemas embebidos donde el control de procesos lo realice el microcontrolador de forma autónoma o se puede decidir realizar estas funciones a través de los dispositivos de red programables de forma centralizada en la nube con el modelo llamado redes inteligentes.

Además, existen otras herramientas como FTTT (If This, Then That) o en español (Si ocurre esto, haz esto otro) el cual es un servicio web que permite crear y programar acciones para automatizar diferentes tareas en Internet, es decir, el control de procesos y almacenamiento de información en la nube.

Son innumerables las tecnologías que ofrecen servicios para el naciente internet de las cosas y aunque coexistan, es difícil saber por dónde empezar, que tecnología utilizar o cual va a predominar. Hoy el internet de las cosas integra los sistemas embebidos, los dispositivos de red programables, las herramientas en la nube y el diseñador decide de qué forma las va a utilizar.

Palabras Claves: Internet de las cosas, IoT en CISCO, Sistemas embebidos, Sistemas operativo IoX.

¹ Msc. en Ingeniería área Telecomunicaciones, Esp. En Telecomunicaciones, Ingeniero Electricista. Docente Investigador GNET. Unidades Tecnológicas de Santander.

Summary: The internet of things seeks to connect people, equipment, devices and everything unimaginable through the internet, for that they were using simple devices like sensors, actuators and microcontrollers. For the actuator to respond in a logical way to a measurement taken by a sensor are used the microncontroladores, which are implemented in electronic boards as Arduino, Edinson and Raspberry.

Currently there are other technologies and network companies, which perform the work of controlling the processes that the microcontroller makes from network devices such as the router or Gateway. While embedded circuit companies are looking for the miniaturization of their cards and ways to make programming easier, network equipment companies like CISCO, enter the internet circle of things with network equipment that have programmable software applications through New operating systems.

In order to realize a project of the internet of things, it is possible to use a design model with embedded systems where the control of processes is carried out by the microcontroller autonomously or it is possible to decide to perform these functions through centrally programmable network devices in The cloud with the model called smart grids.

In addition, there are other tools like FTTT (If This, Then That) or in Spanish (if this happens, do this another) which is a web service that allows you to create and program actions to automate different tasks on the Internet, Processes and storage of information in the cloud.

There are innumerable technologies that offer services for the nascent internet of things and although they coexist, it is difficult to know where to start, which technology to use or which will predominate. Today the internet of things integrates embedded systems, programmable network devices, tools in the cloud and the designer decides how to use them.

Keywords: Internet of things, IoT in CISCO, IOx operating system, embedded systems, Internet of Everything.

1. INTRODUCCIÓN

Las placas de circuitos embebidos como Arduino, Raspberry o el circuito ESP8266, poseen un microcontrolador que toma la información de cualquier cosa o del entorno a través de sensores conectados a sus pines de entrada, analizan y ejecutan una respuesta a través de dispositivos llamados actuadores. Estos dispositivos junto con otros conectados a la nube pueden guardar los datos, compararlos, analizarlos, cambiar las variables del sistema, predecir comportamiento, generar una respuesta automática programada o realizar mensajes través de una red social, email o una llamada telefónica desde o a cualquier lugar del mundo. Todo esto es lo que se llama el Internet de las cosas (Internet of things IoT).

La empresa CISCO tiene una nueva propuesta con su sistema operativo IOx que busca con un dispositivo de red como un gateway o router y conjunto de herramientas de desarrollo de software (software development kit, SDK), tomar la información de los sensores y transmitir su posterior respuesta a los actuadores.

En proyectos de IoT (Internet de las Cosas), el procesado de datos se puede hacer desde tarjetas embebidas, dispositivos de red, un Smartphone, un computador personal, desde la nube utilizando cloud computing o desde la niebla con el modelo de Fog Computing. Para que un proyecto pertenezca al IoT debe estar conectado a internet y para esto se debe garantizar el transporte y manejo de información que en algunos casos el modelo de cloud computing no lo ofrece. Esta es la razón por la cual algunos proponen una capa anterior en este modelo jerárquico de redes llamado la computación en la niebla (Fog Computing) o computación en el borde (computación Edge), creando una infraestructura informática más cercana al perímetro de la red en la cual el procesamiento de los datos de sensores y actuadores junto con las aplicaciones programables, se concentran en los dispositivos al borde de la red, en lugar de trabajar de forma centralizada y tener que viajar al exterior a través de la nube. [1]

La empresa CISCO entre otras, a través de un nuevo sistema operativo llamado IoX permite la eliminación en gran parte de los sistemas embebidos ya que la lógica programable para el control de procesos se haría desde el Gateway de la red y el análisis de datos desde la computación en la niebla lo cual permite que estos datos puedan ser procesados localmente en lugar de hacerlo desde los dispositivos embebidos o ser enviados a la nube o para ello. La computación en la niebla, creado para el Internet de las cosas, da la posibilidad a todos los dispositivos no solo de conectarse a Internet sino aumentando la probabilidad de funcionamiento aun cuando se pierden las conexiones de red.

Se puede afirmar que los procesos del IoT no solo se pueden realizar desde los circuitos embebidos o desde los dispositivos de la red sino que existen herramientas en IFTTT, Amazon o Azure que fácilmente lo pueden hacer. En este caso el control de procesos y análisis de datos se hace totalmente a través de cloud computing.

Es importante para el diseño de proyectos del IoT, determinar de qué forma se realizará el control de procesos y el análisis de datos, teniendo en cuenta que puede realizarse de forma autónoma en cada uno de los circuitos embebidos, a través de dispositivos de red programables utilizando Fog computing o totalmente centralizado con cloud computing.

2. METODOLOGÍA

El primer paso en el diseño de sistemas del internet de las cosas es la determinación de los elementos a utilizar dependiendo de las necesidades del proyecto, las especificaciones de los dispositivos y las limitaciones de la tecnología.

En este caso se ha planteado un proyecto del internet de las cosas a varios grupos de estudiantes de Ingeniería con el rol de diseñadores para la solución de un problema de ciudades inteligentes. Cada uno de ellos está orientado de tal forma que los proyectos se realicen con herramientas de diferentes características. Así, unos grupos están orientados a trabajar en gran parte con diferentes sistemas embebidos, otros con dispositivos de red y los demás con herramientas web como IFTTT. La contrastación de los proyectos se realiza a partir de variables como facilidad en su programación, costos, movilidad o portabilidad, facilidad en la implementación de sistemas de predicción y simulación. La recolección de información se realiza a través del análisis de las conclusiones de cada uno de los proyectos realizados por los grupos de trabajo.

Los indicadores para la evaluación de los proyectos se basaron en estudios anteriores que describen los diferentes aspectos técnicos necesarios para la implementación del internet de las cosas en Colombia, teniendo en cuenta que existen hoy en día oportunidades de investigación y negocios aplicados a esta área. Cabe resaltar que generar aplicaciones IoT móviles de bajo costo, la creación y diseño de dispositivos terminales con sensores integrados se hace un campo interesante por explorar a nivel comercial. [3]

3. RESULTADOS Y DISCUSIÓN.

Cada uno de los modelos del IoT utiliza para control de los procesos, almacenamiento y análisis de datos un lenguaje de programación. El lenguaje de programación difiere dependiendo del fabricante. Empresas como Raspberry utiliza el lenguaje de programación python, otras como Arduino el lenguaje de programación processing y así, existen muchos más. Con resultados en el diseño de proyectos de IoT se toma como no relevante el lenguaje de programación utilizado ya que se puede interactuar fácilmente con cualquiera de ellos. Por ejemplo, en dispositivos con IOx de CISCO se pueden realizar proyectos de IoT utilizando cualquiera de los lenguajes de programación como: JavaScript, Python o Visual.

La programación en JavaScript en dispositivos programables de CISCO es muy parecida a la programación con processing utilizada en dispositivos Arduino.

Figura 1. Lenguajes de programación en IoT.

ARDUINO	PACKET TRACER
<pre>int led = 13; void setup() { pinMode(led, OUTPUT); } void loop() { digitalWrite(led, HIGH); delay(1000); digitalWrite(led, LOW); delay(1000); }</pre>	<pre>function setup() { pinMode(1, OUTPUT); } function loop() { Serial.println("Blinking"); digitalWrite(1, HIGH); delay(1000); digitalWrite(1, LOW); delay(500); }</pre>

Fuente. Autor del artículo. IDE de Arduino y Simulador Packet Tracer

Como se puede apreciar en la figura 1, un programa básico para encender y apagar un led deja en evidencia la similitud en la forma de programación utilizando el lenguaje processing de Arduino y el JavaScript en el simulador Packet Tracer de CISCO. Otras empresas utilizan lenguajes de programación como C, Basic o Assembler.

Una ventaja para un diseñador del IoT a la hora de escoger un modelo de trabajo y los dispositivos a emplear, es el lenguaje de programación con el que más está familiarizado, pero esto no será relevante si este modelo o dispositivo permite programar en cualquier tipo de lenguaje.

Otro aspecto importante a tener en cuenta es el tipo de comunicación empleado entre los diferentes elementos del proyecto IoT ya que los dispositivos electrónicos, sensores y actuadores generalmente debe ser localizado o llevarse en personas, animales o plantas sin realizar ningún tipo de afectación a ellas. Para esta característica, llamada portabilidad, hay que tener en cuenta el peso del dispositivo, las baterías la movilidad del sujeto, la velocidad de los datos que llevarán la información entre otras cosas.

Existen tecnologías inalámbricas como la conocida M-Health (la práctica de la medicina y la salud pública soportada por dispositivos móviles) y los Weareables (objetos de uso diario que llevamos siempre encima a los que se les ha incorporado un microprocesador) que nos acercan al no afectamiento del sujeto que porta los dispositivos del IoT. Hay que tener en cuenta que estos dispositivos en la actualidad su gran mayoría se conectan a internet a través de Smartphone es decir a distancias muy cortas. [3]

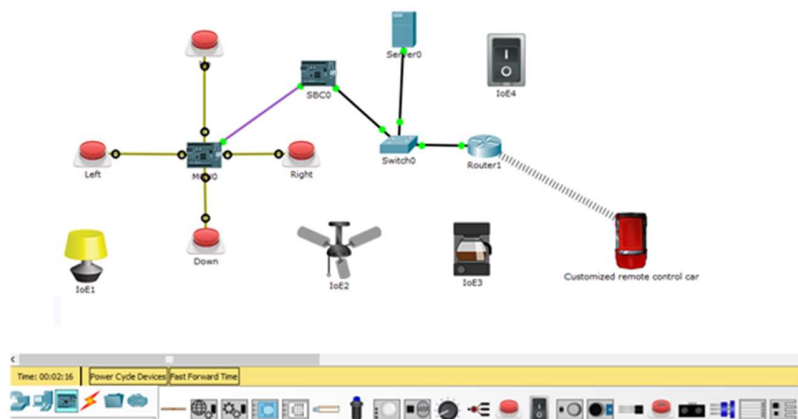
Con la implementación de un sistema operativo como el IOx de CISCO sobre un Gateway que soporte una aplicación programable evitará la utilización del circuito embebido con microcontrolador, la necesidad de un Smartphone para su conexión a internet y de esta forma disminuya elementos en el sistema. En este caso utilizando sensores y actuadores

inalámbricos se tendrá una mayor movilidad y menos afectación que con el sistema tradicional que utiliza circuitos embebidos. También hay que recordar que los sensores consumen menor potencia que los circuitos embebidos y las baterías utilizadas serían más pequeñas.

Como un siguiente punto de comparación, la simulación de un sistema es importante por muchos aspectos. Si se desea capacitar a personal que en un futuro utilice esta tecnología, es más económico y fácil utilizando un simulador. Si se piensa en realizar pruebas para determinar el rendimiento o mejorar el proyecto es útil un simulador. Cada uno de los modelos estudiados tiene un simulador que sirvió para la presentación de resultados en los diferentes proyectos realizados en IoT.

El Packet Tracer permite crear un sistema IoT conformado por varios sensores y actuadores que trabajen de acuerdo a una programación y con las entradas y salidas que se desee como se puede observar en la siguiente figura.

Fig. 2. Internet de las Cosas con Packet tracer



Fuente: Simulación autor del artículo en Packet Tracer de CISCO.

A diferencia de simuladores electrónicos como Proteus o Autodesk Circuit, el Packet Tracer integra los sistemas embebidos o tarjetas programables a los equipos de red como router, switch y otros. Es decir, que se puede diseñar la topología de una red LAN y WAN, asignar direccionamiento IPv4 o IPv6, configurar protocolos de enrutamiento para examinar el comportamiento del IoT de extremo a extremo de la red. Este simulador incorpora dispositivos de red como router de borde o gateway con el nuevo sistema operativo IOX que pueden, además de sus funciones de dispositivos de red, realizar el trabajo de analizar y procesar datos de acuerdo a una programación preestablecida.

Los simuladores de circuitos electrónicos como proteus y Autodesk circuit permiten escoger circuitos embebidos y adicionar actuadores y sensores pero no poseen elementos de red para su implementación y simulación. Por el momento desde la simulación existe una gran brecha entre los diferentes modelos que se analizan para la elaboración de proyectos de IoT. Con el modelo de cloud computing la simulación es más complicada ya que los diferentes sistemas para el análisis de datos y control de procesos localizados en la nube necesitan y esperan los datos tomados por sensores que hasta el momento ningún software de simulación lo realiza.

En un análisis de costos de los diferentes modelos de trabajo en este tipo de proyectos, es sorprendente la disminución en el precio de las tarjetas con sistemas embebidos diseñados para IoT. Existen integrados o tarjetas de desarrollo como el ESP8266 o el ESP32, que poseen un microcontrolador y conexión WIFI o Bluetooth con costos de aproximadamente 5 dólares., esto presenta una ventaja sobre el modelo del internet de la cosas sobre dispositivos de red o con el modelo de cloud computing.

La incorporación de sistemas de predicción en proyectos de IoT se desarrolla más fácilmente a través de aplicaciones en la nube utilizando en este caso el modelo cloud como Big Data, pero los precios son elevados en la actualidad.

4. CONCLUSIONES

En el diseño de proyectos del internet de las cosas existen diferentes modelos dependiendo del sitio donde se realice los diferentes procesos a los datos suministrados por los sensores del sistema. Las especificaciones como portabilidad y seguridad son manejadas por el modelo de redes inteligentes de una forma más eficiente que el modelo con sistemas embebidos.

Actualmente el desarrollo de proyectos de IoT con todo el procesado de datos a través de circuitos embebidos presenta ventajas a la hora de presentarse fallas en la red que incomunican el sistema con la nube, además presenta precios muy bajos en comparación al modelo de redes inteligentes que por tiene Gateway con altos precios.

La similitud de los dos modelos en los lenguajes de programación, los protocolos de comunicación utilizados, el uso de sensores y actuadores hacen que sea prácticamente igual trabajar en cualquiera de los dos modelos.

Como en la mayoría de sistemas, incluyendo los de telecomunicaciones, se busca migrar a sistemas centralizados, por esta razón se establece que poco a poco el sistema de redes inteligentes ira ganando espacio con sus dispositivos de red programables sobre los sistemas tradicionales con circuitos embebidos.

En la transición de un modelo a otro se crea la computación osmótica que reúne la computación en la nube, la computación en la niebla y el desarrollo de aplicaciones desde los circuitos embebidos. La computación osmótica plantea que el desarrollo de aplicaciones, la programación, la seguridad y gestión se maneje en diferentes proporciones desde los dispositivos embebidos, la computación en la niebla y en la computación en la nube, cada una de estas capas permea a las otras dependiendo el tipo de proyecto que se realice.

5. REFERENCIAS

- [1] Bonomi, F., Milito, R., Natarajan, P., & Zhu, J. (2014). Fog computing: A platform for internet of things and analytics. In *Big Data and Internet of Things: A Roadmap for Smart Environments* (pp. 169-186). Springer International Publishing.
- [2] Colle, R. (2017). CMD9 Algoritmos, grandes datos e inteligencia en la red. Una visión crítica. *Revista Mediterránea de Comunicación/Mediterranean Journal of Communication*. (pp.6-9).
- [2] García, L., & Carlos, L. Estudio del impacto técnico y económico de la transición de internet al internet de las cosas (IoT) para el caso colombiano (Doctoral dissertation, Universidad Nacional de Colombia). (pag-74)
- [3] Want, R., Schilit, B. N., & Jenson, S. (2015). Enabling the internet of things. *Computer*, 48(1), 28-35.
- [4] Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future generation computer systems*, 29(7), 1645-1660.
- [5] Stojmenovic, I., & Wen, S. (2014, September). The fog computing paradigm: Scenarios and security issues. In *Computer Science and Information Systems (FedCSIS), 2014 Federated Conference on* (pp. 1-8). IEEE.
- [6] Yi, S., Li, C., & Li, Q. (2015, June). A survey of fog computing: concepts, applications and issues. In *Proceedings of the 2015 Workshop on Mobile Big Data* (pp. 37-42). ACM.
- [7] Bonomi, F., Milito, R., Zhu, J., & Addepalli, S. (2012, August). Fog computing and its role in the internet of things. In *Proceedings of the first edition of the MCC workshop on Mobile cloud computing* (pp. 13-16). ACM.
- [8] Dastjerdi, A. V., Gupta, H., Calheiros, R. N., Ghosh, S. K., & Buyya, R. (2016). Fog computing: Principles, architectures, and applications. *arXiv preprint arXiv:1601.02752*.
- [9] Nardelli, M., Nastic, S., Dustdar, S., Villari, M., & Ranjan, R. (2017). Osmotic Flow: Osmotic Computing+ IoT Workflow. *IEEE Cloud Computing*, 4(2), 68-75.
- [10] Madumal, M. P., Atukorale, D. A. S., & Usoof, T. M. H. A. (2016, September). Adaptive event tree-based hybrid CEP computational model for Fog computing architecture.

In Advances in ICT for Emerging Regions (ICTer), 2016 Sixteenth International Conference on (pp. 5-12). IEEE.