La IoT utiliza cuatro modelos de comunicación comunes:

* Dispositivo a dispositivo
* Dispositivo a nube
* Dispositivo a puerta de enlace (gateway)
* Compartir datos de back-end.

El tipo de tecnología inalámbrica implementada dependerá del modelo de comunicación.

**Device to Device**utiliza Bluetooth, Z-Wave o Zigbee ya que implica la transmisión de pequeñas cantidades de datos.

**Device to Cloud** utiliza tecnología WiFi o celular. Las conexiones en la nube permiten a los usuarios obtener acceso al dispositivo de forma remota.

**Device to Gateway** utiliza la red de su dispositivo inteligente como un teléfono inteligente o un reloj inteligente. Ejemplos de esto son los rastreadores de fitness que cargan datos en su aplicación móvil.

**Backend Data Sharing** extiende el dispositivo único a las comunicaciones en nube a terceros autorizados. Esto puede utilizar cualquier conectividad de red como WiFi, celular o incluso por satélite. Todo se reduce al caso de uso de su negocio

Ventajas de las comunicaciones Wireless:

* Escalable: Las redes inalámbricas no requieren ninguna instalación de hardware. Típicamente involucran configuraciones y pueden estar listos y funcionando en poco tiempo. También se pueden ampliar muy fácilmente sin tener en cuenta las obstrucciones de la instalación. Las tecnologías inalámbricas más recientes utilizan plug and play, incluida la detección automática que ayuda a reducir los tiempos de instalación.
* Bajo coste: Debido al avance en la tecnología inalámbrica, así como al número de fabricantes, el coste de la tecnología inalámbrica ha ido disminuyendo en los últimos años. Además, la mayoría de los sensores inalámbricos vienen con nodos que se pueden ampliar añadiendo nodos adicionales según los requisitos.

Desventajas de las comunicaciones Wireless:

* Interferencia: Los dispositivos electrónicos en las proximidades de las redes inalámbricas pueden interferir fácilmente y pueden causar pérdidas en la conexión o reducir la calidad de la misma. Esto puede conducir a la pérdida de productividad hasta que el problema se identifique y se solucione.
* Velocidad más lenta: Cuando se trata de datos en tiempo real, es imperativo que los datos se transmitan y estén disponibles lo más rápido posible. Las redes inalámbricas son susceptibles a una mayor latencia e interferencia de señal que afecta a la velocidad y consistencia de los datos.

**ZigBee**

ZigBee es una tecnología inalámbrica más centrada en aplicaciones domóticas e industriales. Los perfiles ZigBee PRO y ZigBee Remote Control (RF4CE) se basan en el protocolo IEEE 802.15.4, una tecnología de red inalámbrica que opera a 2,4GHz en aplicaciones que requieren comunicaciones con baja tasa de envío de datos dentro de áreas delimitadas con un alcance de 100 metros, como viviendas o edificios.

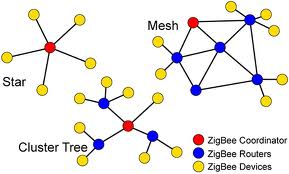
IEEE 802.15.4 es un estándar que define el nivel físico y el control de acceso al medio de redes inalámbricas de área personal con tasas bajas de transmisión de datos (low-rate wireless personal area network, LR-WPAN). El grupo de trabajo [IEEE 802.15](https://es.wikipedia.org/wiki/IEEE_802.15) es el responsable de su desarrollo. También es la base sobre la que se define la especificación de [ZigBee](https://es.wikipedia.org/wiki/ZigBee), cuyo propósito es ofrecer una solución completa para este tipo de redes construyendo los niveles superiores de la pila de protocolos que el estándar no cubre.

ZigBee/RF4CE tiene algunas ventajas significativas como el bajo consumo en sistemas complejos, seguridad superior, robustez, alta escalabilidad y capacidad para soportar un gran número de nodos. Así, es una tecnología bien posicionada para marcar el camino del control wireless y las redes de sensores en aplicaciones IoT y M2M.

* Estándar: ZigBee 3.0 basado en IEEE 802.15.4
* Frecuencia: 2.4GHz
* Alcance: 10-100m
* Velocidad de transferencia: 250kbps

**XBee**

es el nombre comercial del Digi de una familia de módulos de comunicación por radio y están basados en el estándar zigbee, pero digi tiene muchos Xbee y algunos son zigbee estándar y otros son propietarios o modificaciones del estándar. Existen muchos módulos Xbee basados en el estándar [IEEE 802.15.4](https://en.wikipedia.org/wiki/IEEE_802.15.4)



Más información: <https://aprendiendoarduino.wordpress.com/2016/11/16/zigbeexbee/>

**WiFi**

Normalmente la conectividad WiFi es la opción obvia elegida por los desarrolladores dada la omnipresencia de WiFi en entornos domésticos y comerciales: existe en la actualidad una extensa infraestructura ya instalada que transfiere datos con rapidez y permite manejar grandes cantidades de datos. Actualmente, el standard WiFi más habitual utilizado en los hogares y en muchas empresas es el 802.11n, ofreciendo un rendimiento significativo en un rango de cientos de megabits por segundo, muy adecuado para la transferencia de archivos, pero que consume demasiada potencia para desarrollar aplicaciones IoT.

* Estándar: Basado en 802.11n
* Frecuencia: 2,4GHz y 5GHz
* Alcance: Aproximadamente 50m
* Velocidad de transferencia: hasta 600 Mbps, pero lo habitual es 150-200Mbps, en función del canal de frecuencia utilizado y del número de antenas (el standard 802.11-ac ofrece desde 500Mbps hasta 1Gbps)

**Bluetooth**

Bluetooth es una de las tecnologías de transmisión de datos de corto alcance más establecidas, muy importante en el ámbito de la electrónica de consumo. Las expectativas apuntan a que será clave para desarrollar dispositivos wearable, ya que permitirá el establecimiento de conexiones IoT, probablemente a través de un smartphone.

El nuevo Bluetooth de baja energía, también conocido como Bluetooth LE o Bluetooth Smart, es otro protocolo importante para desarrollar aplicaciones IoT. Se caracteriza por ofrecer un alcance similar al de la tecnología Bluetooth normal pero con un consumo de energía significativamente reducido.

Es importante destacar que la versión 4.2, gracias a la incorporación del Internet Protocol Support Profile, permite conectarse directamente a internet mediante IPv6/6LoWPAN. Esto facilita el utilizar la infraestructura IP existente para gestionar dispositivos Bluetooth Smart basado en “edge computing”.

* Estándar: Bluetooth 4.2
* Frecuencia: 2,4GHz (ISM)
* Alcance: 50-150m (Smart/LE)
* Velocidad de transferencia: 1Mbps (Smart/LE)

**Thread**

En la actualidad, el protocolo de red más innovador basado en IPv6 es Thread. Diseñado para domótica, está basado en 6LowPAN, y del mismo modo que aquel, no es un protocolo de aplicaciones IoT como Bluetooth o ZigBee. Se diseñó como un complemento WiFi, puesto que aunque la tecnología Wi-Fi funciona muy bien en dispositivos de consumo, tiene limitaciones al utilizar en configuraciones de domótica.

Lanzado a mediados del 2014 por Thread Group, este protocolo sin canon de uso se basa en varios protocolos como IEEE 802.15.4, IPv6 y 6LoWPAN.

Es una solución resistente basada en IP para aplicaciones IoT.

Diseñado para trabajar sobre chips IEEE 802.15.4 ya existentes de fabricantes como Freescale y Silicon Labs, Thread es compatible con redes de topología de malla al utilizar radio transceptores IEEE802.15.4, siendo capaz de manejar hasta 250 nodos con altos niveles de autenticación y cifrado.

Una actualización de software relativamente sencilla permite a los usuarios utilizar thread en dispositivos ya compatibles con IEEE 802.15.4.

* Estándar: Thread, basado en IEEE802.15.4 y 6LowPAN
* Frecuencia: 2,4GHz (ISM)
* Alcance: N/A
* Velocidad de transferencia: N/A

**Red de telefonía móvil**

Cualquier aplicación IoT que necesite funcionar en grandes áreas puede beneficiarse de las ventajas de la comunicación móvil GSM/3G/4G.

La red de telefonía móvil es capaz de enviar grandes cantidades de datos, especialmente a través de 4G, aunque el consumo de energía y el coste económico de la conexión podrían ser demasiado altos para muchas aplicaciones.

Sin embargo, puede ser ideal para proyectos que integren sensores y que no requieran un ancho de banda muy grande para enviar datos por Internet.

* Estándares: GSM/GPRS/EDGE (2G), UMTS/HSPA (3G), LTE (4G)
* Frecuencias: 900 / 1800 / 1900 / 2100
* Alcance: hasta 35km para GSM; hasta 200km para HSPA
* Velocidad de transferencia (descarga habitual): 35-170kps (GPRS), 120-384kbps (EDGE), 384Kbps-2Mbps (UMTS), 600kbps-10Mbps (HSPA), 3-10Mbps (LTE)

**Hologram**

Conectividad celular para IoT

Web: <https://hologram.io/>

HW compatible de Hologram: <https://hologram.io/hologram-compatible-hardware/>

Permite una sim gratuita para probar con 1MB mensual

Pricing: <https://hologram.io/pricing/>

**Neul**

El concepto de este sistema es similar al de Sigfox y funciona en la banda sub-1GHz. Neul aprovecha pequeños fragmentos de la “banda blanca” de las estaciones de TV para ofrecer alta escabilidad, amplia cobertura y bajo costes.

Este sistema se basa en el chip Iceni, que se comunica utilizando los “banda blanca” de la radio para acceder al espectro UHF de alta calidad. Ya está disponible debido a la transición analógica a la televisión digital.

La tecnología de comunicaciones que utiliza se llama Weightless, que es una nueva tecnología de red inalámbrica ampliada diseñada para aplicaciones IoT que compite contra las soluciones GPRS, 3G, CDMA y LTE WAN.

La velocidad de transferencia de datos puede ir de unos bits por segundo hasta 100 Mbps en el mismo enlace. Desde el punto de vista del consumo, los dispositivos consumen tan solo de 20 a 30 mA, es decir, de 10 a 15 años de autonomía con 2 pilas AA.

Para poder emplear esta tecnología hay que tener en cuenta la decisión que se haya tomado acerca del uso de las frecuencias de la banda blanda.

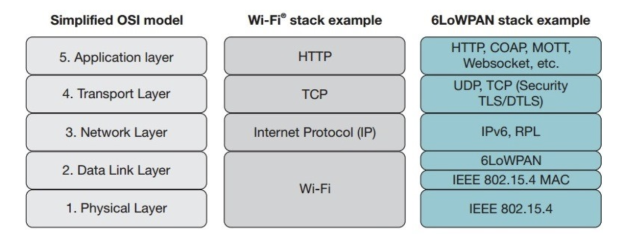
* Estándar: Neul
* Frecuencia: 900MHz (ISM), 458MHz (UK), 470-790MHz (espacios en blanco)
* Alcance: 10km
* Velocidad de transferencia: Desde unos pocos bps hasta 100kbps

**6LoWPAN**

**6LoWPAN** (IPv6 over Low power Wireless Personal Area Networks) es un estándar que posibilita el uso de [IPv6](https://es.wikipedia.org/wiki/IPv6) sobre redes basadas en el estándar IEEE 802.15.4 (LoRa, zigbee, etc…). Hace posible que dispositivos como los nodos de una red inalámbrica puedan comunicarse directamente con otros dispositivos IP.

6LowPAN es una tecnología inalámbrica basada en IP. En vez de tratarse de una tecnología de protocolos de aplicaciones IoT, como Bluetooth o ZigBee, 6LowPAN es un protocolo de red que permite mecanismos de encapsulado y compresión de cabeceras. Esta tecnología ofrece libertad de banda de frecuencia y capa física, por lo que se puede utilizar a través de múltiples plataformas de comunicaciones, como Ethernet, Wi-Fi, 802.15.4 y sub-1GHz ISM.

Capas de red:



6LoWPAN:

* <https://en.wikipedia.org/wiki/6LoWPAN>
* Protocolo Thread: <https://en.wikipedia.org/wiki/Thread_(network_protocol)>
* <http://www.ti.com/lsds/ti/wireless_connectivity/6lowpan/overview.page>
* <https://www.lsr.com/white-papers/zigbee-vs-6lowpan-for-sensor-networks>

6LowPAN (IPv6 Low-power wireless Personal Area Network) es una tecnología inalámbrica basada en IP muy importante. En vez de tratarse de una tecnología de protocolos de aplicaciones IoT, como Bluetooth o ZigBee, 6LowPAN es un protocolo de red que permite mecanismos de encapsulado y compresión de cabeceras. Esta tecnología ofrece libertad de banda de frecuencia y capa física, por lo que se puede utilizar a través de múltiples plataformas de comunicaciones, como Ethernet, Wi-Fi, 802.15.4 y sub-1GHz ISM.

Una característica clave es la introducción de la pila IPv6 (protocolo de internet versión 6), una innovación clave en el avance de IoT en los últimos años, ya que con IPv6 se ofrecen aproximadamente 5 x 10E28 direcciones IP a nivel global, permitiendo que cualquier objeto o dispositivo embebido tenga su propia dirección IP única para conectarse a Internet.

Ha sido diseñada especialmente para el hogar y la automatización de edificios proporcionando un mecanismo de transporte básico para producir sistemas de control complejos e interconexión de dispositivos de un modo económico a través de una red inalámbrica de bajo consumo.

Diseñada para enviar paquetes IPv6 sobre redes IEEE 802.15.4, para luego implementar protocolos superiores como TCP, UDP, HTTP, COAP, MQTT y websockets, 6LowPAN es una red de topología en malla robusta, escalable y auto-regenerativa. Los routers pueden encaminar datos enviados a otros dispositivos, mientras que los hosts permanecen inactivos mucho tiempo.

* Estándar: RFC6282
* Frecuencia: adaptable a múltiples capas físicas como Bluetooth Smart (2.4GHz), ZigBee o comunicación RF de bajo consumo (sub-1GHz)
* Alcance: N/A
* Velocidad de transferencia: N/A

Hardware 6LoWPAN: <https://zolertia.io/>

**LoRaWAN**

Es una especificación de una red LPWAN (Low Power Wide Area Network) propuesta por la [LoRa Alliance](https://www.lora-alliance.org/) y pensada para comunicar dispositivos de bajo coste y bajo consumo alimentados por baterías. La especificación cubre las capas PHY y MAC de la red, dejando a las aplicaciones el resto de capas. En la banda ISM de 868MHz (915 MHz en otras regiones), con un bitrate de hasta decenas de kbps (de 0.3 kbps hasta  50 kbps).

Enlaces:

* <https://es.wikipedia.org/wiki/LoRaWAN>
* <http://www.rfwireless-world.com/Terminology/LoRa-technology-basics.html>
* <http://blog.iot-partners.com/tecnologia/lorawan-una-red-de-largo-alcance-para-iot/>
* Moteino con LoRa: <https://lowpowerlab.com/guide/moteino/lora-support/>
* <http://www.st.com/en/wireless-connectivity/lorawan.html?querycriteria=productId=SC2150>

Esta tecnología se parece en algunos aspectos a Sigfox y a Neul. LoRaWAN está diseñada para implementar redes de área amplia (WAN) con características específicas para soportar comunicaciones móviles, bidireccionales, económicas y seguras para aplicaciones de IoT, M2M, ciudades inteligentes y aplicaciones industriales.

Optimizada para bajo consumo de energía y para ofrecer amplias redes con millones y millones de dispositivos, sus velocidades de transferencia de datos van desde 0,3 kbps hasta 50 kbps.

* Estándar: LoRaWAN
* Frecuencia: Varias
* Alcance: 2-5km (entorno urbano), 15km (entorno rural)
* Velocidad de transferencia: 0,3-50 kbps.

**Z-Wave**

Z-Wave es una tecnología RF de bajo consumo diseñada inicialmente para productos de domótica como controladores de iluminación y sensores. Optimizado para la comunicación fiable de baja latencia de pequeños paquetes de datos, alcanza velocidades de datos de hasta 100kbit/s, opera en la banda de sub-1 GHz y es robusta frente a interferencias de Wi-Fi y otras tecnologías inalámbricas en el rango 2,4 GHz como Bluetooth o ZigBee. Es totalmente compatible con redes de topología de malla, no necesita un nodo coordinador y es muy escalable, permitiendo controlar hasta 232 dispositivos.

Z-Wave utiliza un protocolo más simple que otras tecnologías lo que permite una mayor rapidez en el desarrollo, pero el único fabricante de chips compatibles es la empresa Sigma Design, en comparación con la multitud de empresas que ofrecen productos de otras tecnologías inalámbricas como ZigBee o Bluetooth.

* Estándar: Z-Wave Alliance ZAD12837 / ITU-T G.9959
* Frecuencia: 900MHz (Banda ISM)
* Alcance: 30m
* Velocidad de transferencia: 9,6/40/100kbit/s

**NFC**

NFC (Near Field Communication) es una tecnología que permite dos vías simultáneas de interacción segura entre dispositivos electrónicos, siendo especialmente adecuada para smartphones, permitiendo a los consumidores realizar transacciones de pago, acceder al contenido digital y conectar dispositivos electrónicos, todo ellos sin contacto. Esencialmente, amplía la capacidad de la tecnología contacless de las tarjetas inteligentes permitiendo conexiones punto a punto y modos de funcionamiento activos y pasivos.

* Estándar: ISO/IEC 18000-3
* Frecuencia: 13.56MHz (ISM)
* Alcance: 10cm
* Velocidad de transf.: 100–420kbps

**nRF24**

Este dispositivo NRF2401, integra en un único chip, toda la electrónica y bloques funcionales precisos, para establecer comunicaciones RF (Radio Frecuencia) entre dos o más puntos a diferentes velocidades, (Hasta 2  Mb/seg) con corrección de errores y protocolo de reenvío cuando es necesario, sin intervención del control externo, lo que nos permite aislarnos de todo el trabajo sucio y complicado relacionado con la transmisión física.

Información de producto: <http://www.nordicsemi.com/eng/Products/2.4GHz-RF/nRF24L01>

**Wize**

La tecnología Wize se basa en la frecuencia de 169 MHz y se ha utilizado durante más de 10 años para la medición inteligente por parte de las compañías de agua y gas. La tecnología ha mostrado un rendimiento excepcional, especialmente en lo que se refiere a la penetración de la radio en interiores. Esto lo hace perfecto para aplicaciones de IoT en entornos urbanos donde las paredes suelen mitigar la propagación de otras comunicaciones de radio.

Características.

* Basado en una norma robusta y fiable EN-13757 – Wireless M-Bus
* Capaz de alcanzar a larga distancia, hasta 20 KM
* Consumo de energía extremadamente bajo: hasta 20 años de duración de la batería con 1 mensaje/día
* Bidireccional con programación por aire (OTA)
* Penetración profunda de la radio en interiores
* Solución flexible: Sin bloqueo de chip, sin bloqueo de telecomunicaciones, posibilidad de utilizar la red existente o crear nuevas redes.

Protocolo: <https://www.allwize.io/wize-protocol>

Wize Alliance: <https://www.wize-alliance.com/>

Más información: <https://www.allwize.io/post/the-wize-protocol-the-new-trendy-iot-standard>

Dispositivo compatible con Arduino: <https://www.kickstarter.com/projects/1230929587/extreme-lpwa-arduino-board-for-iot-using-the-wize?lang=es>

**Sigfox**

Es una solución de conectividad celular mundial para el Internet of Things pensada para comunicaciones de baja velocidad que permite reducir los precios y el consumo de energía para los dispositivos conectados. La solución de conectividad SIGFOX se basa en una infraestructura de antenas y de estaciones de base totalmente independientes de las redes existentes.

En la red SIGFOX se transmiten mensajes de 12 bytes, pudiendo enviar 140 mensajes al día.

Sigfox trabaja con fabricantes como Texas Instruments, Atmel, Silicon Labs y otros para poder ofrecer distintos tipos de SOC, transceptores y componentes de conexión a su red. En el caso de smartphones y tablets, actualmente no son compatibles con esta red, pero, al no tener licencia de uso, su inclusión sería realmente económica y sencilla.

* <http://www.nobbot.com/redes/sigfox-la-red-para-el-internet-de-las-cosas/>
* <https://about.sofia2.com/2014/11/26/que-es-sigfox/>
* <https://en.wikipedia.org/wiki/Sigfox>
* <https://www.sigfox.com/es>
* Módulo: <https://www.cooking-hacks.com/sigfox-module-for-arduino-waspmote-raspberry-pi-intel-galileo-868-mhz-7184>
* Shield: <http://snootlab.com/lang-en/retired-products/547-akeru-beta-32-en.html>

Es una alternativa de amplio alcance es Sigfox, que en términos de alcance está entre Wi-Fi y la comunicación móvil. Utiliza bandas ISM, que se pueden utilizar sin necesidad de adquirir licencias.

Sigfox responde a las necesidades de muchas aplicaciones M2M que funcionan con una batería pequeña y solo requieren niveles menores de transferencia de datos, allí donde WiFi se queda demasiado corto y la comunicación móvil es muy cara y consume demasiada energía.

Sigfox utiliza una tecnología llamada Ultra Narrow Band (UNB) diseñada para funcionar con bajas velocidades de transferencias de 10 a 1.000 bits por segundo.

Solo consume 50 microvatios (la comunicación móvil consume 5.000 microvatios) además de poder mantenerse en stand-by 20 años con una batería 2.5Ah (0,2 años para comunicaciones móviles).

Esta tecnología es robusta, energéticamente eficiente y funciona como una red escalable que puede comunicarse con millones de dispositivos móviles a lo largo de muchos kilómetros cuadrados. Así pues, es adecuada para aplicaciones M2M como: contadores inteligentes, monitores médicos, dispositivos de seguridad, alumbrado público y sensores ambientales.

El sistema Sigfox utiliza los transceptores inalámbricos que funcionan en la banda sub-1GHz ofreciendo un rendimiento excepcional, mayor alcance y un consumo mínimo.

* Estándar: Sigfox
* Frecuencia: 900MHz
* Alcance: 30-50km (ambientes rurales), 3-10km (ambientes urbanos)
* Velocidad de transferencia: 10-1000bps

Más información: <https://www.rs-online.com/designspark/eleven-internet-of-things-iot-protocols-you-need-to-know-about>

Esta entrada se publicó en [Arduino](https://aprendiendoarduino.wordpress.com/category/arduino/), [Conectividad IoT](https://aprendiendoarduino.wordpress.com/category/conectividad-iot/), [Curso IoT Open Source](https://aprendiendoarduino.wordpress.com/category/curso-iot-open-source/), [LoRa](https://aprendiendoarduino.wordpress.com/category/lora/), [LoRaWAN](https://aprendiendoarduino.wordpress.com/category/lorawan/), [LPWAN](https://aprendiendoarduino.wordpress.com/category/lpwan/), [Sigfox](https://aprendiendoarduino.wordpress.com/category/sigfox/) y está etiquetada con [Arduino](https://aprendiendoarduino.wordpress.com/tag/arduino/), [Bluetooth](https://aprendiendoarduino.wordpress.com/tag/bluetooth/), [Comunicaciones](https://aprendiendoarduino.wordpress.com/tag/comunicaciones/), [Comunicaciones IoT](https://aprendiendoarduino.wordpress.com/tag/comunicaciones-iot/), [Curso IoT Open Source](https://aprendiendoarduino.wordpress.com/tag/curso-iot-open-source/), [Ethernet](https://aprendiendoarduino.wordpress.com/tag/ethernet/), [LoRa](https://aprendiendoarduino.wordpress.com/tag/lora/), [LoRaWAN](https://aprendiendoarduino.wordpress.com/tag/lorawan/), [Redes](https://aprendiendoarduino.wordpress.com/tag/redes/), [RS485](https://aprendiendoarduino.wordpress.com/tag/rs485/), [Sigfox](https://aprendiendoarduino.wordpress.com/tag/sigfox/), [Wifi](https://aprendiendoarduino.wordpress.com/tag/wifi/), [Xbee](https://aprendiendoarduino.wordpress.com/tag/xbee/) en [19 octubre, 2019](https://aprendiendoarduino.wordpress.com/2019/10/19/conectividad-iot-2/).

[Redes LPWAN/LoRa](https://aprendiendoarduino.wordpress.com/2018/11/21/redes-lpwan-lora/)

[2 respuestas](https://aprendiendoarduino.wordpress.com/2018/11/21/redes-lpwan-lora/#comments)

**Redes LPWAN:**[**https://www.aprendiendoarduino.com/2018/03/05/redes-lpwan/**](https://www.aprendiendoarduino.com/2018/03/05/redes-lpwan/)

**Arduino y LoRaWAN:**[**https://www.aprendiendoarduino.com/2018/03/07/arduino-y-lorawan/**](https://www.aprendiendoarduino.com/2018/03/07/arduino-y-lorawan/)

**Demo LoRa con Moteino:**[**https://www.aprendiendoarduino.com/2018/03/07/demo-lora-con-moteino/**](https://www.aprendiendoarduino.com/2018/03/07/demo-lora-con-moteino/)

Esta entrada se publicó en [Arduino](https://aprendiendoarduino.wordpress.com/category/arduino/), [Curso IoT Open Source](https://aprendiendoarduino.wordpress.com/category/curso-iot-open-source/), [Gateway](https://aprendiendoarduino.wordpress.com/category/gateway/), [LoRa](https://aprendiendoarduino.wordpress.com/category/lora/), [LoRaWAN](https://aprendiendoarduino.wordpress.com/category/lorawan/), [LPWAN](https://aprendiendoarduino.wordpress.com/category/lpwan/), [moteino](https://aprendiendoarduino.wordpress.com/category/moteino/) y está etiquetada con [Arduino](https://aprendiendoarduino.wordpress.com/tag/arduino/), [Curso IoT Open Source](https://aprendiendoarduino.wordpress.com/tag/curso-iot-open-source/), [LoRa](https://aprendiendoarduino.wordpress.com/tag/lora/), [LoRa Gateway](https://aprendiendoarduino.wordpress.com/tag/lora-gateway/), [LoRaWAN](https://aprendiendoarduino.wordpress.com/tag/lorawan/), [LPWAN](https://aprendiendoarduino.wordpress.com/tag/lpwan/), [moteino](https://aprendiendoarduino.wordpress.com/tag/moteino/) en [21 noviembre, 2018](https://aprendiendoarduino.wordpress.com/2018/11/21/redes-lpwan-lora/).

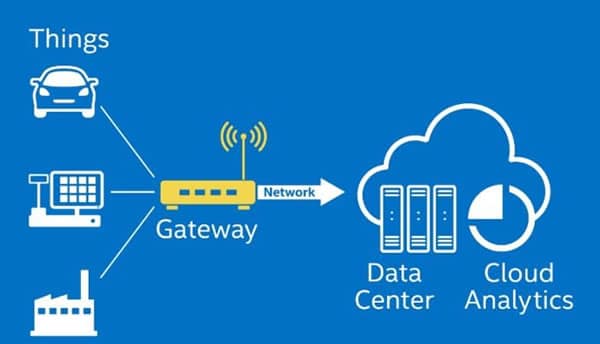
[Gateways IoT](https://aprendiendoarduino.wordpress.com/2018/11/21/gateways-iot/)

[1 respuesta](https://aprendiendoarduino.wordpress.com/2018/11/21/gateways-iot/#comments)

Un Gateway IoT es un dispositivo físico o un programa de software que sirve como punto de conexión entre la nube y los controladores, sensores y dispositivos inteligentes. Todos los datos que se mueven a la nube, o viceversa, pasan por el gateway, que puede ser un dispositivo de hardware dedicado o un programa de software. Un gateway IoT también puede denominarse pasarela inteligente o nivel de control.

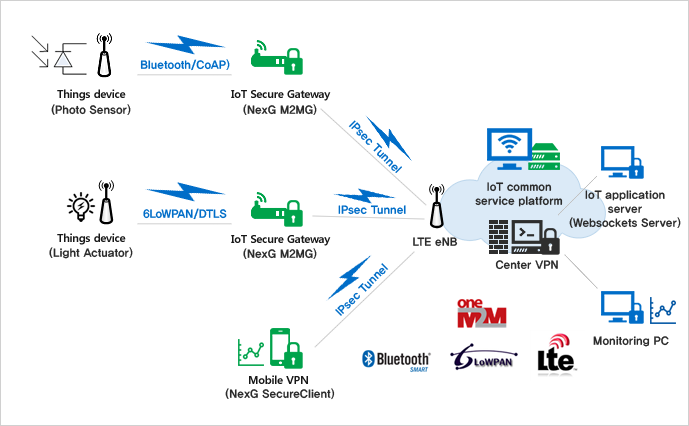
Algunos sensores generan decenas de miles de puntos de datos por segundo. Una pasarela proporciona un lugar para preprocesar esos datos localmente en el borde antes de enviarlos a la nube. Cuando los datos se agregan, se resumen y se analizan tácticamente en el borde, se minimiza el volumen de datos que deben ser enviados a la nube, lo que puede tener un gran impacto en los tiempos de respuesta y en los costes de transmisión de la red.

Otra ventaja de una pasarela de IoT es que puede proporcionar seguridad adicional para la red de IoT y los datos que transporta. Dado que el gateway gestiona información que se mueve en ambas direcciones, puede proteger los datos que se mueven hacia la nube de fugas y dispositivos de IoT de ser comprometidos por ataques externos maliciosos con características tales como detección de manipulaciones, cifrado, generadores de números aleatorios de hardware y motores de cifrado.



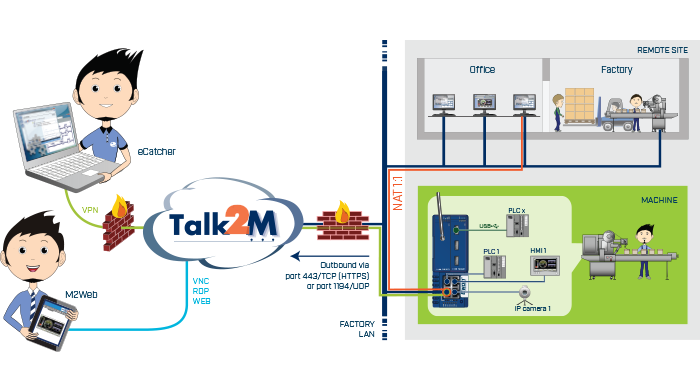
La pasarela IoT desempeña un papel importante en la gestión de los dispositivos. Cada dispositivo (sensor/actuador) tiene un caso de uso diferente y emite mensajes a través de diferentes canales como Wifi, BLE, Zigbee, Ethernet, RF, LPWAN, LTE, etc. y el gateway realiza varias funciones como conectividad de dispositivos, traducción de protocolos, agregación, filtrado, correlación, seguridad, actualizaciones, administración y más. Se sitúa entre los dispositivos y la plataforma de nube.

Además un Gateway puede hacer conexión VPN segura entre localizaciones diferentes, permitiendo unir de forma segura diferentes puntos a través de Internet.



Routers y Gateways industriales inteligentes <https://ewon.biz/es>

Hace no tantos años la conexión remota era por módem o por GSM con conexiones pto a pto. En la actualidad usamos internet para el telecontrol, pero es un problema el tema de la seguridad. Ya existen dispositivos como el router industrial eWON Flexy <https://ewon.biz/es/productos/flexy> modular que es servidor OPC UA y cliente openvpn, es muy potente para conectar y dar funcionalidades adicionales a unos autómatas.



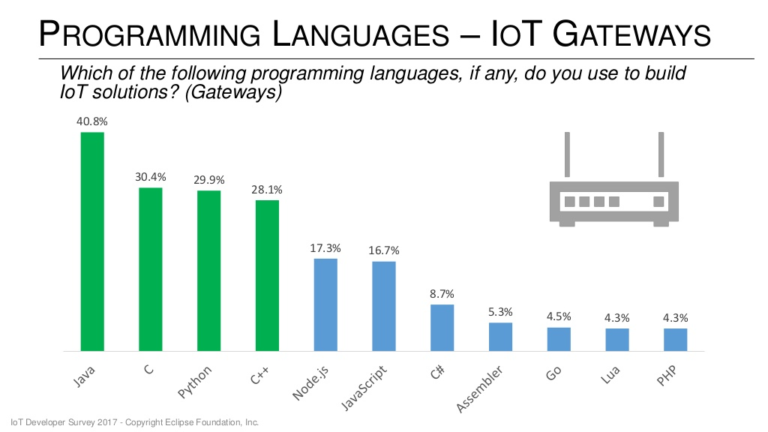
eWON monta la VPN y al ser servidor modbus TCP y OPC UA, es posible acceder remotamente y de forma segura a los datos del autómata e integrarlo con datos de otras localizaciones.

Ejemplo Gateway LoRa:



Y el código: <https://github.com/jecrespo/aprendiendoarduino-lora/blob/master/Demo_LoRa/rf95_server/rf95_server.ino>

Programación de los Gateways IoT:



Esta entrada se publicó en [Arduino](https://aprendiendoarduino.wordpress.com/category/arduino/), [Comunicación](https://aprendiendoarduino.wordpress.com/category/comunicacion/), [Conectividad IoT](https://aprendiendoarduino.wordpress.com/category/conectividad-iot/), [Curso IoT Open Source](https://aprendiendoarduino.wordpress.com/category/curso-iot-open-source/), [Gateway](https://aprendiendoarduino.wordpress.com/category/gateway/), [LoRa](https://aprendiendoarduino.wordpress.com/category/lora/) y está etiquetada con [Arduino](https://aprendiendoarduino.wordpress.com/tag/arduino/), [Comunicación Inalámbrica](https://aprendiendoarduino.wordpress.com/tag/comunicacion-inalambrica/), [Curso IoT Open Source](https://aprendiendoarduino.wordpress.com/tag/curso-iot-open-source/), [eWON](https://aprendiendoarduino.wordpress.com/tag/ewon/), [Gateway IoT](https://aprendiendoarduino.wordpress.com/tag/gateway-iot/), [LoRa](https://aprendiendoarduino.wordpress.com/tag/lora/) en [21 noviembre, 2018](https://aprendiendoarduino.wordpress.com/2018/11/21/gateways-iot/).

[Material Curso IoT](https://aprendiendoarduino.wordpress.com/2018/11/02/material-curso-iot/)

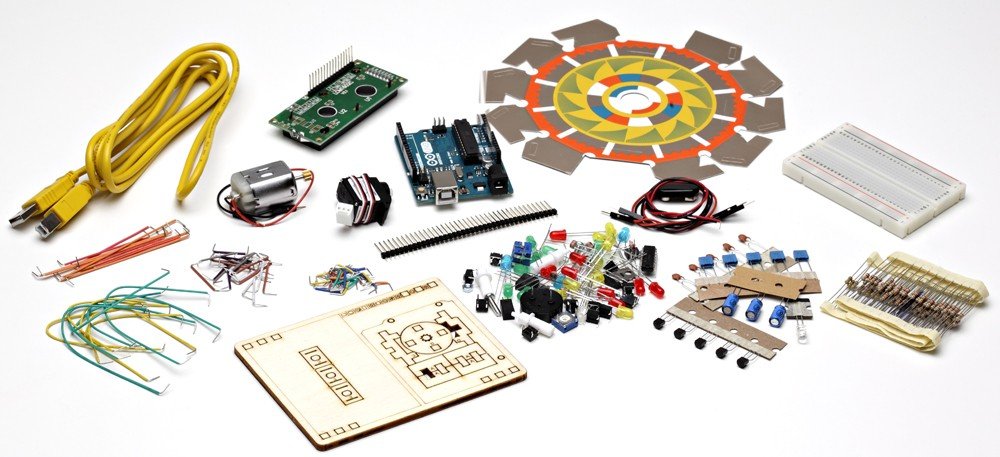
[1 respuesta](https://aprendiendoarduino.wordpress.com/2018/11/02/material-curso-iot/#comments)

Para hacer las prácticas y simular un ecosistema de IoT disponemos de hardware barato y muy accesible para cualquiera para poder utilizar en una prueba de concepto.

El material es en préstamo y se firmará una hoja de entrega de material.

20x Arduino Starter Kit (<https://www.arduino.cc/en/Main/ArduinoStarterKit>) que contiene un Arduino Uno, una serie de sensores y actuadores y diversos elementos electrónicos necesarios para realizar las prácticas y proyectos. Componentes:

* 1 [Arduino / Genuino Uno](https://www.arduino.cc/en/Main/ArduinoBoardUno)
* 1 [USB cable](https://www.arduino.cc/documents/datasheets/USBcable.PDF)
* 1 Breadboard 400 points
* 70 Solid core jumper wires
* 1 Easy-to-assemble wooden base,
* 1 [9v battery snap](https://www.arduino.cc/documents/datasheets/9vBatterySnap.PDF)
* 1 Stranded jumper wires (black)
* 1 Stranded jumper wires (red)
* 6 [**Photoresistor [VT90N2 LDR]**](https://www.arduino.cc/documents/datasheets/LDR-VT90N2.pdf)
* 3 [**Potentiometer 10kOhms**](https://www.arduino.cc/documents/datasheets/Potentiometer.PDF) y [**potentiometer**](http://arduino.cc/documents/datasheets/ACP_potentiometers.pdf)
* 10 [**Pushbuttons**](https://www.arduino.cc/documents/datasheets/Button.pdf)
* 1 [**Temperature sensor [TMP36]**](https://www.arduino.cc/en/uploads/Main/TemperatureSensor.pdf)
* 1 [**Tilt sensor**](https://www.arduino.cc/documents/datasheets/TiltSensor.PDF)
* 1 [**alphanumeric LCD (16×2 characters)**](https://www.arduino.cc/documents/datasheets/LCDscreen.PDF)
* 1 [LED (bright white)](https://www.arduino.cc/documents/datasheets/LED(white).pdf)
* 1 [LED (RGB)](https://www.arduino.cc/documents/datasheets/LED(RGB).pdf)
* 8 [LEDs (red)](https://www.arduino.cc/documents/datasheets/LED(red).pdf)
* 8 [LEDs (green)](https://www.arduino.cc/documents/datasheets/Leds(Green).pdf)
* 8 [LEDs (yellow)](https://www.arduino.cc/documents/datasheets/LEDY-L-7113YT.pdf)
* 3 [LEDs (blue)](https://www.arduino.cc/documents/datasheets/LED(blue).pdf)
* 1 [**Small DC motor 6/9V**](https://www.arduino.cc/documents/datasheets/DCmotor.PDF)
* 1[**Small servo motor**](https://www.arduino.cc/documents/datasheets/servoMotor.PDF)
* 1 [**Piezo capsule [PKM17EPP-4001-B0]**](https://www.arduino.cc/documents/datasheets/piezoCapsule.pdf) y [**piezo**](https://www.arduino.cc/documents/datasheets/PIEZO-PKM22EPPH4001-BO.pdf)
* 1 [**H-bridge motor driver [L293D]**](https://www.arduino.cc/documents/datasheets/H-bridge_motor_driver.PDF)
* 1 [**Optocouplers [4N35]**](https://www.arduino.cc/documents/datasheets/Optocouplers.pdf)
* 2 [Mosfet transistors [IRF520]](https://www.arduino.cc/documents/datasheets/MosfetTransistor.pdf)
* 5 [Capacitors 100uF](https://www.arduino.cc/documents/datasheets/cap100uF.pdf)
* 5 [Diodes [1N4007]](https://www.arduino.cc/documents/datasheets/Diodes.pdf)
* 3 [Transparent gels (red, green, blue)](https://www.arduino.cc/documents/datasheets/TrasparentGels.pdf)
* 1 [Male pins strip (40×1)](https://www.arduino.cc/documents/datasheets/Strip40x1.pdf)
* 20 [Resistors 220 Ohms](https://www.arduino.cc/documents/datasheets/Resistors.pdf)
* 5 [Resistors 560 Ohms](https://www.arduino.cc/documents/datasheets/Resistors.pdf)
* 5 [Resistors 1 kOhms](https://www.arduino.cc/documents/datasheets/Resistors.pdf)
* 5 [Resistors 4.7 kOhms](https://www.arduino.cc/documents/datasheets/Resistors.pdf)
* 20 [Resistors 10 kOhms](https://www.arduino.cc/documents/datasheets/Resistors.pdf)
* 5 [Resistors 1 MOhms](https://www.arduino.cc/documents/datasheets/Resistors.pdf)
* 5 [Resistors 10 MOhms](https://www.arduino.cc/documents/datasheets/Resistors.pdf)



10x Kit XBee( <http://www.digikey.es/product-detail/en/digi-international/XKB2-AT-WWG/602-1550-ND/5271212>) para desarrollo de aplicaciones con XBee. Datasheet: <https://media.digikey.com/pdf/Data%20Sheets/Digi%20International%20PDFs/XBee_Arduino_Compatible_Coding_Platform.pdf>. Compuesto por:

* Arduino Leonardo-compatible
* XBee shield XBee 802.15.4 modules
* XBee breadboard adapters
* XBee USB adapter
* One 2-axis joysticks
* Six pushbuttons
* Solderless breadboards
* Breadboard power supplies
* 18 LEDs
* 10kΩ potentiometers
* Package of 330Ω resistors
* 9 V batteries and battery clips
* Micro USB cable
* Mini USB cable
* Bundle – jumper wires



20x Kits Raspberry Pi compuestos por:

* Raspberry Pi 3 Model B <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>
* Carcasa Raspberry Pi
* Tarjeta microSD
* Cable alimentación Raspberry Pi
* Cable VGA para Raspberry Pi
* Pilas 9V recargables
* Cargadores de pilas 9V

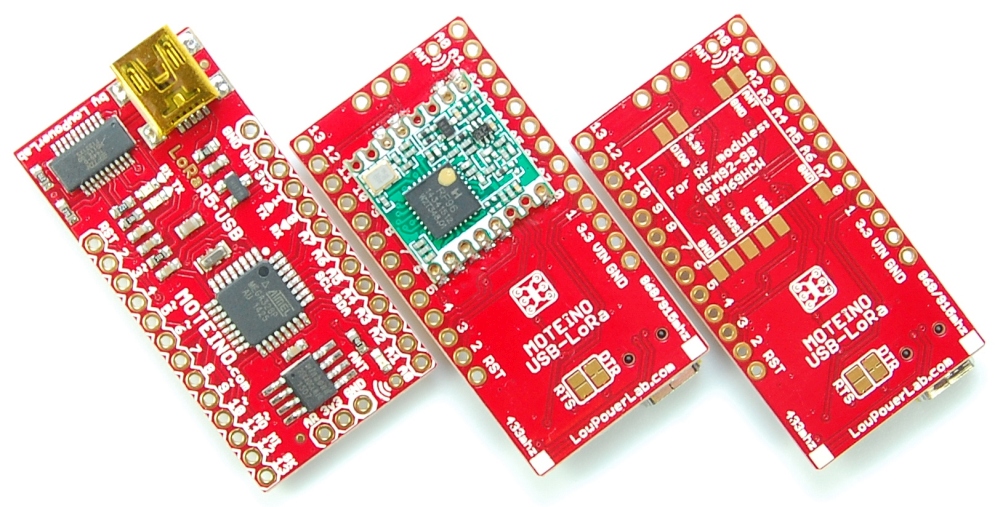
Kit Wemos D1 mini

* 20x wemos D1 mini <https://www.amazon.es/AZDelivery-D1-Mini-desarrollo-compatible/dp/B0754W6Z2F>
* 20x cables USB – micro usb
* 20x shields para wemos D1 mini relé <https://www.amazon.es/Ils-Piezas-Escudo-Versi%C3%B3-ESP8266/dp/B07F6D4SC9/>
* 20x shields para wemos D1 mini neopixel <https://diyprojects.io/deal-shield-point-led-driver-ws2812b-wemos-d1-mini-compatible-adafruit-neopixel/>
* 20x shields para wemos D1 mini oled <https://www.amazon.es/Ils-Piezas-Shield-V2-0-0-Botones/dp/B07FM8HH4K/>



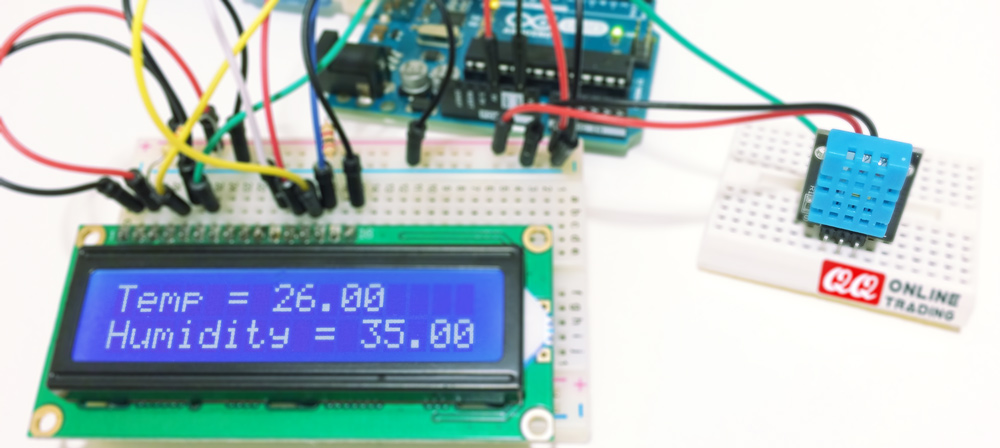
Módulos de comunicación:

* 20x Ethernet Shield: <https://www.arduino.cc/en/Main/ArduinoEthernetShield>
* 20x moteinos con comunicación LoRa <https://lowpowerlab.com/shop/product/100> y Transceiver LoRa RFM95
* 20x Módulo Ultra low power 2.4GHz RF nRF24L01+: <https://www.seeedstudio.com/item_detail.html?p_id=1394>
* 20x Módulo Wifi ESP8266: <https://www.seeedstudio.com/WiFi-Serial-Transceiver-Module-w%26-ESP8266-1MB-Flash-p-2485.html>
* 20x Módulo bluetooth HC-05: <http://www.prometec.net/producto/modulo-bluetooth-hc-05/>



Otros módulos:

* 20x sondas de temperatura y humedad DHT11 <https://www.amazon.es/WINGONEER-humedad-temperatura-frambuesa-Arduino/dp/B06XHJ1BPC>
* 20x pantalla LCD I2C [https://es.aliexpress.com/store/product/1602-LCD-Module-Blue-Yellow-Green-Screen-with-IIC-I2C-16×2-LCD-Backlight-Module-LCD-1602/1326062\_32891917063.html](https://es.aliexpress.com/store/product/1602-LCD-Module-Blue-Yellow-Green-Screen-with-IIC-I2C-16x2-LCD-Backlight-Module-LCD-1602/1326062_32891917063.html)
* 20x Módulo IMU MPU6050 <http://www.miniinthebox.com/es/la-ultima-mpu-6050-modulo-6000-de-6-ejes-giroscopio-acelerometro_p4348915.html>
* 20x Programador FTDI: <http://www.tinyosshop.com/index.php?route=product/product&product_id=600>
* 20x Conversor analógico digital MCP3008 <https://cdn-shop.adafruit.com/datasheets/MCP3008.pdf>



Esta entrada se publicó en [Arduino](https://aprendiendoarduino.wordpress.com/category/arduino/), [Curso IoT Open Source](https://aprendiendoarduino.wordpress.com/category/curso-iot-open-source/), [ESP8266](https://aprendiendoarduino.wordpress.com/category/esp8266/), [Kit Arduino](https://aprendiendoarduino.wordpress.com/category/kit-arduino/), [LoRa](https://aprendiendoarduino.wordpress.com/category/lora/), [moteino](https://aprendiendoarduino.wordpress.com/category/moteino/), [Práctica](https://aprendiendoarduino.wordpress.com/category/practica/), [Raspberry Pi](https://aprendiendoarduino.wordpress.com/category/raspberry-pi/), [Wemos D1 mini](https://aprendiendoarduino.wordpress.com/tag/wemos-d1-mini/), [Xbee](https://aprendiendoarduino.wordpress.com/category/xbee/) y está etiquetada con [Arduino](https://aprendiendoarduino.wordpress.com/tag/arduino/), [Arduino Kit](https://aprendiendoarduino.wordpress.com/tag/arduino-kit/), [Curso IoT Open Source](https://aprendiendoarduino.wordpress.com/tag/curso-iot-open-source/), [DHT11](https://aprendiendoarduino.wordpress.com/tag/dht11/), [ESP8266](https://aprendiendoarduino.wordpress.com/tag/esp8266/), [LoRa](https://aprendiendoarduino.wordpress.com/tag/lora/), [moteino](https://aprendiendoarduino.wordpress.com/tag/moteino/), [Pantalla LCD](https://aprendiendoarduino.wordpress.com/tag/pantalla-lcd/), [Raspberry Pi](https://aprendiendoarduino.wordpress.com/tag/raspberry-pi/), [Wemos D1 mini](https://aprendiendoarduino.wordpress.com/tag/wemos-d1-mini/), [Xbee](https://aprendiendoarduino.wordpress.com/tag/xbee/) en [2 noviembre, 2018](https://aprendiendoarduino.wordpress.com/2018/11/02/material-curso-iot/).

[Arduino y LoRaWAN](https://aprendiendoarduino.wordpress.com/2018/03/07/arduino-y-lorawan/)

[3 respuestas](https://aprendiendoarduino.wordpress.com/2018/03/07/arduino-y-lorawan/#comments)

Arduino MKR WAN 1300

MKR WAN 1300 es una placa potente que combina la funcionalidad de la conectividad MKR Zero y LoRa. Es la solución ideal para los fabricantes que desean diseñar proyectos de IoT con una mínima experiencia previa en redes que tengan un dispositivo de baja potencia.

La placa MKR WAN 1300 tiene comunicación inalámbrica unido a un diseño de la placa [MKR Zero Board](https://www.hwlibre.com/mkrzero-una-nueva-placa-arduino-los-proyectos-educativos/), es decir, que tendremos soporte para aplicaciones de 32 bits. La placa cuenta con 256KB de Memoria flash y 32KB SRAM. Puede funcionar con la energía de dos pilas de 1,5V y todo en un tamaño de 67,64 x 25mm. Al tener comunicación inalámbrica, el dispositivo al que se conecte tendrá opción de comunicarse a Internet.

El MKR WAN 1300 usar el módulo Murata CMWX1ZZABZ Lo-Ra module que lleva el transceiver Semtech SX1276:

* <https://www.murata.com/products/lpwa/lora>
* <https://wireless.murata.com/eng/products/rf-modules-1/lpwa/type-abz.html>
* Datasheet: <https://wireless.murata.com/datasheet?/RFM/data/type_abz.pdf>

Más información: <https://store.arduino.cc/mkr-wan-1300>



**Buen artículo para sobre el MKR 1300:**[**http://tinkerman.cat/arduino-mkr-wan-1300/**](http://tinkerman.cat/arduino-mkr-wan-1300/)

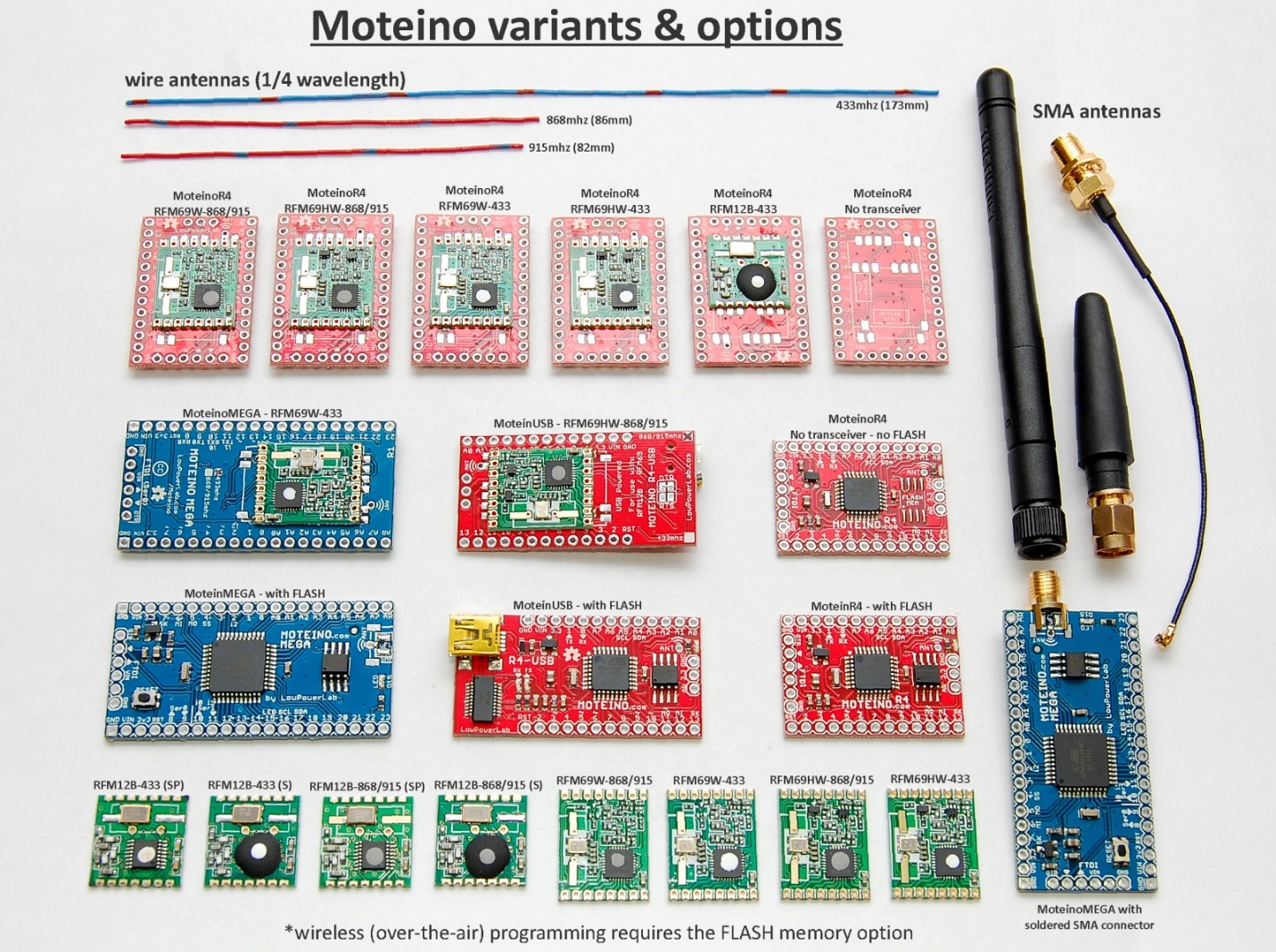
Moteino

Moteino es una plataforma de desarrollo compatible con Arduino inalámbrica de baja potencia basada en el popular chip ATmega328p utilizado en el Arduino-UNO, lo que lo hace 100% compatible con el IDE de Arduino (entorno de programación).

Para la programación, necesitará un [adaptador FTDI](https://lowpowerlab.com/shop/ftdi-adapter) externo para cargar los sketchs, con las ventajas de un menor costo y un tamaño más pequeño. La variante MoteinoUSB incluye el convertidor de serie USB.

Los Moteinos son compatibles y se pueden comunicar con cualquier otra plataforma Arduino o de desarrollo que utilice los populares transceptores HopeRF RFM69 o LoRa. Moteino también viene con un chip de memoria flash SPI opcional para programación inalámbrica o registro de datos.

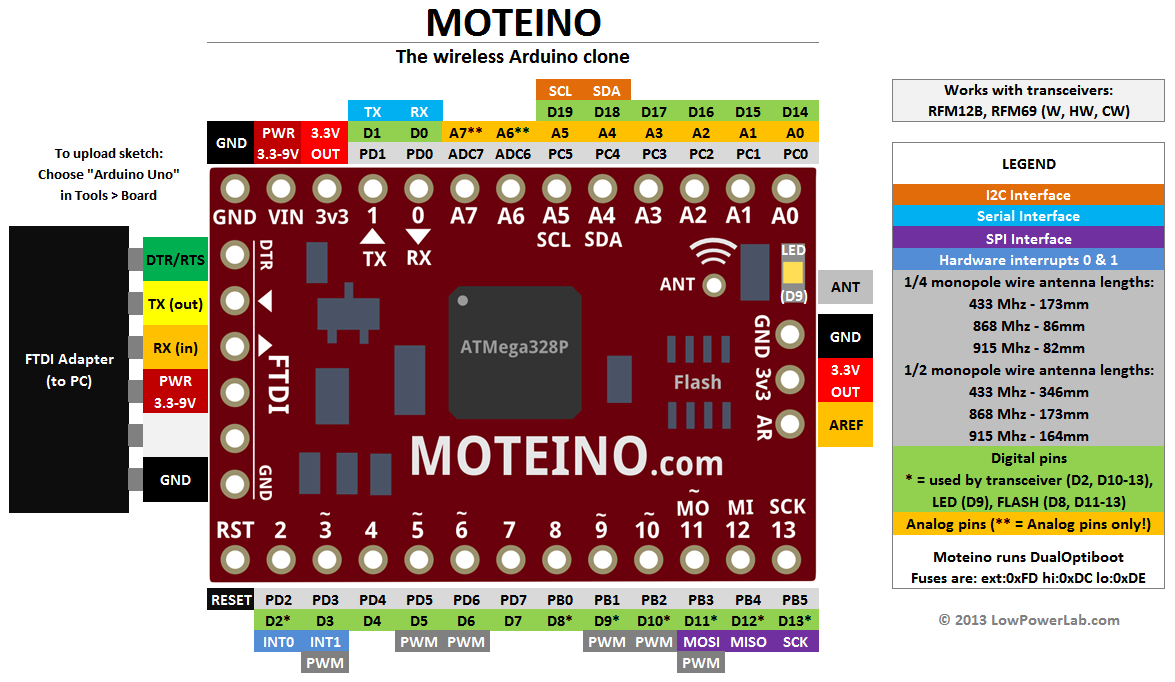
Web Moteino: <https://lowpowerlab.com/guide/moteino/>



Moteino fue diseñado para ser una plataforma de desarrollo compacta, altamente personalizable y asequible, adecuada para IoT, domótica y proyectos inalámbricos de largo alcance. Estas son algunas de las características que distinguen a Moteino:

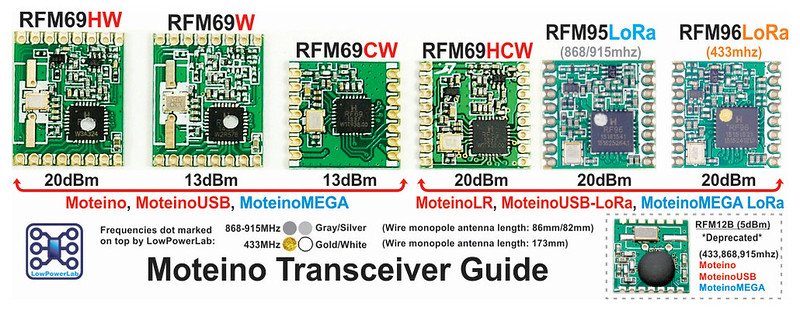
* diseño modular pequeño y ligero que se adapta a recintos minúsculos
* las configuraciones flexibles permiten el uso de varios transceptores inalámbricos
* potencia realmente ultra baja: con tan solo ~ 2uA alcanzables en el modo de suspensión profunda, los Moteinos permiten que los proyectos con batería, como los sensores inalámbricos de movimiento/entorno, funcionen durante años. El modo de suspensión de Watchdog está en ~ 6uA (activación periódica). El nuevo [8Mhz Moteino](https://lowpowerlab.com/guide/moteino/moteino-8mhz/) permite el modo de sueño 100nA más bajo posible
* Las radios sub-Ghz y LoRa producen un rango mucho más largo que las bandas de 2.4Ghz
* [programable de forma inalámbrica](https://lowpowerlab.com/guide/moteino/wireless-programming/): puede volver a flashearlo sin cables, cuando se implementa en ubicaciones difíciles (solo con radios RFM69)
* fácil de usar desde el familiar IDE Arduino, muchos ejemplos de código brindados para ayudarlo a comenzar

Pinout:



Los transceiver soportados por Moteino son:

* <https://lowpowerlab.com/guide/moteino/transceivers/>
* <https://lowpowerlab.com/shop/category/73>



Transceiver Datasheets

* <http://www.hoperf.com/upload/rf/RFM69W-V1.3.pdf>
* <http://www.hoperf.com/upload/rf/RFM69HW-V1.3.pdf>
* <http://www.hoperf.com/upload/rf/RFM95_96_97_98W.pdf>
* <http://www.hoperf.com/upload/rf/rfm12b.pdf>

Muy buena explicación de los módulos de Adafruit: <https://learn.adafruit.com/adafruit-rfm69hcw-and-rfm96-rfm95-rfm98-lora-packet-padio-breakouts/overview>

Librería para los módulos RFM69: <https://github.com/LowPowerLab/RFM69>

**IMPORTANTE**: Los módulos RFM69 no son LoRa y no son compatibles con los módulos RFM95/RFM96. Además los módulos RFM95/RFM96 necesitan de una librería de terceros.

Más información RFM69:

* <https://www.sparkfun.com/products/12775>
* <https://lowpowerlab.com/2015/11/11/rfm69_atc-automatic-transmission-control/>
* <https://es.rs-online.com/web/p/modulos-rf-de-baja-potencia/7931995/>
* <https://www.digitalsmarties.net/products/rfm-board>
* <https://www.digitalsmarties.net/products/rfm69cw-868-mhz>

Uso con Lora: <https://lowpowerlab.com/guide/moteino/lora-support/>

Librería para los módulos LoRa RFM95 (868-915mhz) and RFM96 (433mhz).: <http://www.airspayce.com/mikem/arduino/RadioHead/index.html>

Getting started para instalar el soporte y las librerías: <https://lowpowerlab.com/guide/moteino/programming-libraries/>

Github: <https://github.com/LowPowerLab> y Librería: <https://github.com/LowPowerLab/Moteino>

Comprar Moteino:<https://lowpowerlab.com/shop/>

Moteino weather shield: <https://lowpowerlab.com/2016/09/09/weathershield-r2-released/> with a BME280 which includes all Temperature/Humidity/Pressure readings all in 1 sensor.

Moteino PowerShield: <https://lowpowerlab.com/guide/powershield/>

Gateway LoRa con Moteino + Raspberry Pi:

* <https://lowpowerlab.com/guide/gateway/>
* Hat: <https://lowpowerlab.com/guide/mightyhat/>

Dragino

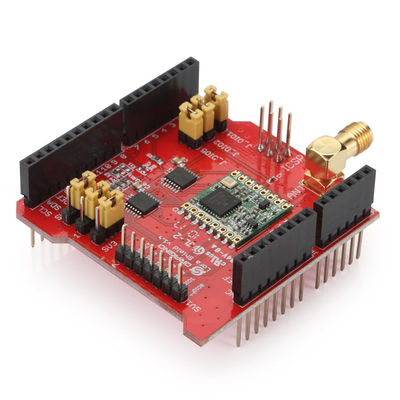
En Dragino <http://www.dragino.com/> podemos encontrar Hardware para LoRa: <http://www.dragino.com/products/products-list.html>

Wiki: <http://wiki.dragino.com/index.php?title=Main_Page>

La librería recomendada es: <https://github.com/matthijskooijman/arduino-lmic>, pero puede usarse la librería Radiohead: <http://www.airspayce.com/mikem/arduino/RadioHead/>

LoRa Shield:

* Shield de LoRa para Arduino: <http://www.dragino.com/products/lora/item/102-lora-shield.html>
* Manual: <http://wiki.dragino.com/index.php?title=Lora_Shield>



Otro HW LoRa compatible con Arduino

Existen más HW de desarrollo compatible con Arduino con módulos LoRa diferentes:

* M5Stack: <http://m5stack.com/>
* ESP32 LoRa: <https://es.aliexpress.com/item/868-MHz-915-MHz-SX1276-ESP32-LoRa-0-96-Pulgadas-de-Pantalla-OLED-Azul-Kit-32/32836591865.html?spm=a219c.12010108.1000023.13.7fdcecc0EA9d6W>
* ESP8266 + LoRa: <https://www.tindie.com/products/electronictrik/esp8266-rfm95-lorawan-node-pcb-with-battery/>
* Módulo Industrial: <https://www.kickstarter.com/projects/1322607643/4zerobox-the-industrial-toolkit-for-iot-solution-p>
* LoPy: <https://pycom.io/product/lopy/>

Gateways LoRa

En las redes LoRaWan un gateway es un dispositivo dentro de la arquitectura de red que recibe los datos transmitidos por un dispositivo de nodo final y que reenvían los paquetes de datos a un servidor de red centralizado. Los datos de un nodo final LoRa pueden ser recibidos por múltiples puertas de enlace (gateway),

Los gateways o puertas de enlace son un puente transparente entre los dispositivos finales y el servidor de red central. Uno o más dispositivos finales se conectan a una o más puertas de enlace, mediante una conexión inalámbrica de un solo salto, usando tecnología RF LoRa™ o FSK, formando así una red en estrella.

Una o más puertas de enlace se conectan al servidor de red central por medio de conexiones IP estándar, formando así una red en estrella. Las comunicaciones entre los dispositivos y el servidor de red, son generalmente unidireccionales o bidireccionales, pero el estándar también soporta multidifusión, permitiendo la actualización de software en forma inalámbrica, u otras formas de distribución de mensajes en masa.

Los gateways son enrutadores equipados con un concentrador LoRa, lo que les permite recibir paquetes LoRa. Por lo general, puede encontrar dos tipos de puertas de enlace:

* Las pasarelas se ejecutan con un firmware mínimo, por lo que son de bajo costo y fáciles de usar (por ejemplo, The Things Gateway) y solo ejecutan el software de reenvío de paquetes.
* Gateways que ejecutan un sistema operativo, para el cual el software de reenvío de paquetes se ejecuta como un programa de fondo (por ejemplo, Kerlink IoT Station, Multitech Conduit). Esto le da más libertad al administrador del gateway para administrar su puerta de enlace e instalar su propio software.

**Una forma de montar un gateway LoRa barato es con una Raspberry Pi y un hat de Moteino con un módulo LoRa:**

* [**https://lowpowerlab.com/guide/gateway/**](https://lowpowerlab.com/guide/gateway/)
* Hat: <https://lowpowerlab.com/guide/mightyhat/>

Un gateway simple con LoPy: <https://www.hackster.io/bucknalla/lopy-lorawan-nano-gateway-using-micropython-and-ttn-a9fb19>

Construir un gateway LoRa barato: <http://cpham.perso.univ-pau.fr/LORA/RPIgateway.html>

Módulos LoRa para conectar un ordenador y haga de gateway: <https://www.cooking-hacks.com/waspmote-gateway-sx1272-lora-sma-4-5-dbi-868-mhz> y tutorial LoRa gateway Libelium: <http://www.libelium.com/development/waspmote/documentation/lora-gateway-tutorial/>

Lista de gateways de loriot: <https://www.loriot.io/lora-gateways.html>

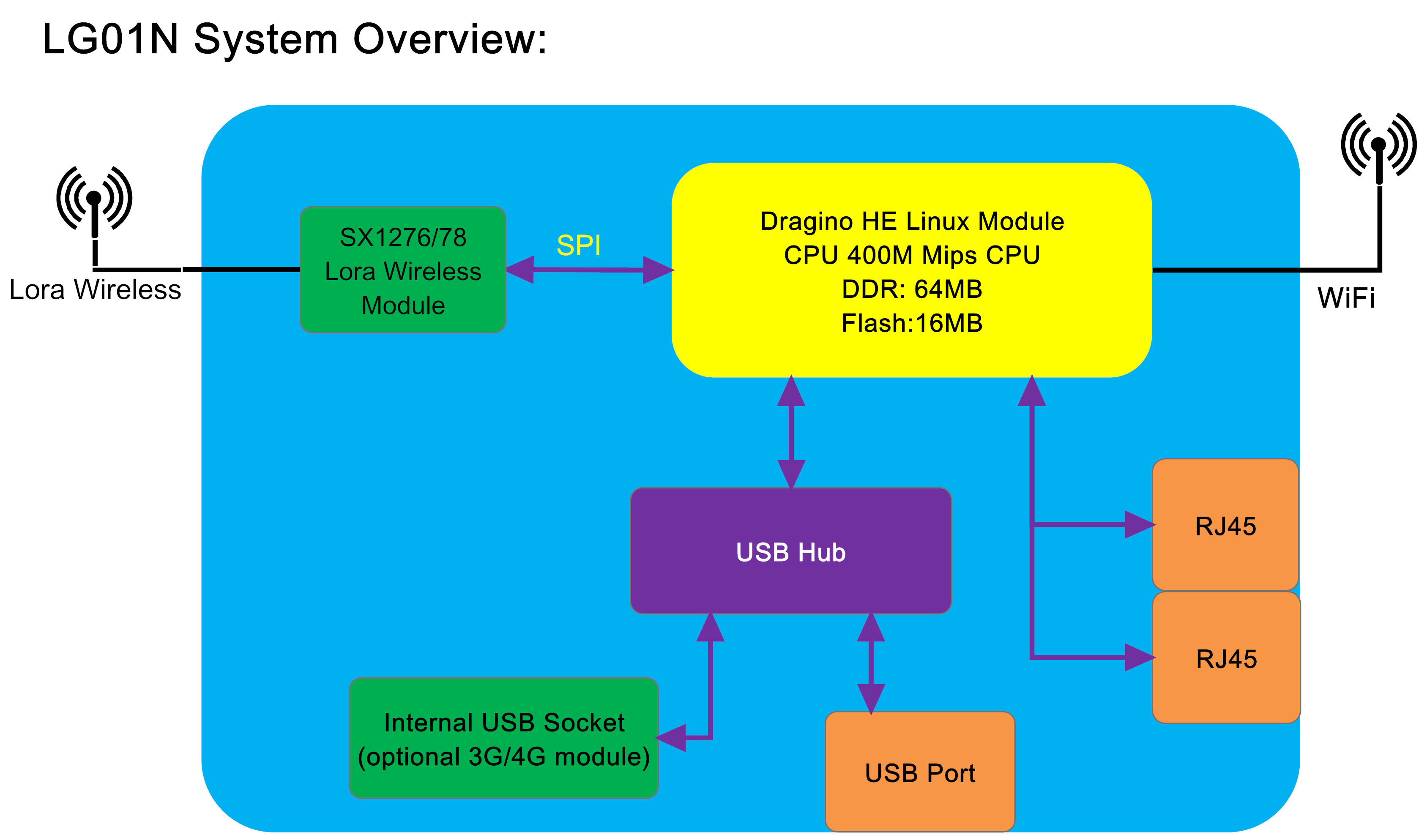
The things gateway: <https://www.thethingsnetwork.org/docs/gateways/gateway/>

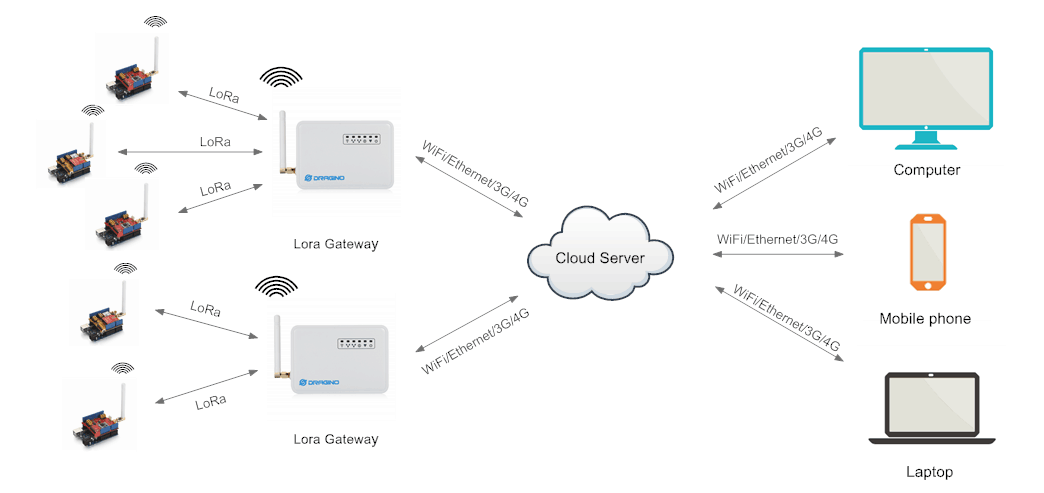
The Things Gateway permite que dispositivos como sensores y computadoras integradas se conecten a internet. Con un proceso fácil de conectar, está creando el aspecto más sustancial de su red de datos IoT. Active la puerta de enlace en solo 5 minutos y cree su propia red local. Con la capacidad de servir a miles de nodos, la puerta de enlace es el componente principal de su red conectada. Esta versión funciona a 868MHz para uso en la UE y 915Mhz para uso en los EE.UU.

Lista de gateways de thethingsnetwork:

* <https://www.thethingsnetwork.org/docs/gateways/start/list.html>
* <https://github.com/TheThingsNetwork/docs/blob/master/_content/gateways/start/list.md>

Gateway draguino (open wrt): <http://www.dragino.com/products/lora/item/119-lg01-s.html>





Ejemplo con Dragino para usarlo como gateway (Lora Shield + Arduino Yun Shield):

* Lora GPS tracker: <http://www.instructables.com/id/Dragino-LoRa-GPS-Tracker-1/>
* LoRa Shield como módulo RFM: <https://www.seeedstudio.com/Dragino-LoRa-Shield-support-433M-frenquency-p-2672.html>

Más información:

* <https://www.thethingsnetwork.org/docs/gateways/>
* Tutorial Libelium: <http://www.libelium.com/development/waspmote/documentation/lora-gateway-tutorial/>
* <https://www.semtech.com/products/wireless-rf/lora-gateways>
* <https://docs.mbed.com/docs/lora-with-mbed/en/latest/intro-to-lora/>
* <https://www.youtube.com/watch?v=ZFVA6cQyheY>

Librería RadioHead LoRa

Una de las librerías más usadas para módulos LoRa con Arduino es RadioHead: <http://www.airspayce.com/mikem/arduino/RadioHead/index.html>

Proporciona una biblioteca completa orientada a objetos para enviar y recibir mensajes paquetizados a través de una variedad de radios de datos comunes y otros transportes para microprocesadores integrados.

RadioHead consta de 2 grupos principales de clases: driversy managers.

* Los drivers proporcionan acceso de bajo nivel a un rango de diferentes radios y otros transportes de mensajes paquetizados.
* Los managers brindan servicios de envío y recepción de mensajes de alto nivel para una variedad de requisitos diferentes.

Cada programa de RadioHead tendrá una instancia de un driver para proporcionar acceso a la radio o transporte de datos, y generalmente un manager que usa ese driver para enviar y recibir mensajes para la aplicación. El programador debe instanciar un driver y un manager e inicializar el manager. A partir de entonces, las funciones del manager se pueden usar para enviar y recibir mensajes.

También es posible usar un driver por sí mismo, sin un manager, aunque esto solo permite un transporte no confiable y sin dirección a través de las funciones del driver.

Se admite una amplia gama de plataformas de microprocesadores.

Unos ejemplos de drivers:

* [RH\_RF69](http://www.airspayce.com/mikem/arduino/RadioHead/classRH__RF69.html) Works with Hope-RF RF69B based radio modules, such as the RFM69 module
* [RH\_NRF24](http://www.airspayce.com/mikem/arduino/RadioHead/classRH__NRF24.html) Works with Nordic nRF24 based 2.4GHz radio modules, such as nRF24L01 and others.
* [RH\_RF95](http://www.airspayce.com/mikem/arduino/RadioHead/classRH__RF95.html) Works with Semtech SX1276/77/78/79, Modtronix inAir4 and inAir9, and HopeRF RFM95/96/97/98 and other similar LoRa capable radios. Supports Long Range (LoRa) with spread spectrum frequency hopping, large payloads etc.
* [RH\_Serial](http://www.airspayce.com/mikem/arduino/RadioHead/classRH__Serial.html) Works with RS232, RS422, RS485, RS488 and other point-to-point and multidropped serial connections, or with TTL serial UARTs such as those on Arduino and many other processors, or with data radios with a serial port interface. [RH\_Serial](http://www.airspayce.com/mikem/arduino/RadioHead/classRH__Serial.html) provides packetization and error detection over any hardware or virtual serial connection. Also builds and runs on Linux and OSX.
* [RHEncryptedDriver](http://www.airspayce.com/mikem/arduino/RadioHead/classRHEncryptedDriver.html) Adds encryption and decryption to any RadioHead transport driver, using any encrpytion cipher supported by ArduinoLibs Cryptogrphic Library <http://rweather.github.io/arduinolibs/crypto.html>

Managers, cualquier manager puede usarse con cualquier driver:

* [RHDatagram](http://www.airspayce.com/mikem/arduino/RadioHead/classRHDatagram.html) Addressed, unreliable variable length messages, with optional broadcast facilities.
* [RHReliableDatagram](http://www.airspayce.com/mikem/arduino/RadioHead/classRHReliableDatagram.html) Addressed, reliable, retransmitted, acknowledged variable length messages.
* [RHRouter](http://www.airspayce.com/mikem/arduino/RadioHead/classRHRouter.html) Multi-hop delivery of RHReliableDatagrams from source node to destination node via 0 or more intermediate nodes, with manual routing.
* [RHMesh](http://www.airspayce.com/mikem/arduino/RadioHead/classRHMesh.html) Multi-hop delivery of RHReliableDatagrams with automatic route discovery and rediscovery.

Esta librería es compatible entre otros con:

* Arduino and the Arduino IDE
* Moteino <http://lowpowerlab.com/moteino/>
* ESP8266 on Arduino IDE and Boards Manager per <https://github.com/esp8266/Arduino>
* ChipKIT Core with Arduino IDE
* Particle Photon <https://store.particle.io/collections/photon>
* ESP32 built using Arduino IDE 1.8.1 or later
* Raspberry Pi Uses BCM2835 library for GPIO <http://www.airspayce.com/mikem/bcm2835/> Currently works only with [RH\_NRF24](http://www.airspayce.com/mikem/arduino/RadioHead/classRH__NRF24.html) driver or other drivers that do not require interrupt support.
* Linux and OSX Using the RHutil/HardwareSerial class, the [RH\_Serial](http://www.airspayce.com/mikem/arduino/RadioHead/classRH__Serial.html) driver and any manager will build and run on Linux and OSX.

Para los módulos con moteino que se ha usado en la demo, son necesarios los drivers: <http://www.airspayce.com/mikem/arduino/RadioHead/classRH__RF95.html>

Si se quiere añadir una capa de seguridad debe usarse la clase: <http://www.airspayce.com/mikem/arduino/RadioHead/classRHEncryptedDriver.html>

Si se quiere usar direccionamiento debe usarse la clase: <http://www.airspayce.com/mikem/arduino/RadioHead/classRHDatagram.html>

**IMPORTANTE PARA MODULOS LORA**, la librería está configurada por defecto a 434: Check if you have set the right frequency:After putting the library in the right place, you have to also modify the frequency to the frequency you want to use, the position of this issetFrequency() in the file: arduino-xxx\libraries\RadioHead\RH\_RF95.cpp;

Para los módulos RFM95 de moteino debe ponerse: setFrequency(868.0);

He hecho un fork de la librería con la modificación para módulos LoRa: <https://github.com/jecrespo/RadioHead>

Proyecto LoRa con Moteino

A la hora de afrontar un proyecto con LoRa para monitorizar un entorno donde no tenemos acceso a una red ethernet/wifi ni toma eléctrica, podemos planteamos usar Moteino como una solución basada en Arduino de bajo consumo y con módulos LoRa integrados.

La primera duda es que módulo de radio o transceiver usar el RFM69 o RFM95:

* RFM69 no es LoRa usa modulación FSK en lugar de la modulación LoRa
* RFM95 es LoRa estándar.

RFM69 y RFM95 son módulos de radio para comunicación a larga distancia, donde la velocidad de transmisión no es crítica (no se hace streaming de vídeo). Al usar modulación diferente no son compatibles entre ellos.

Estos módulos de radio vienen en cuatro variantes (dos tipos de modulación y dos frecuencias). Los RFM69 son los más fáciles de usar, y son bien conocidos y entendidos. Las radios LoRa son más potentes, pero también más caros.

Comparativa y explicación de los módulos: <https://learn.adafruit.com/adafruit-rfm69hcw-and-rfm96-rfm95-rfm98-lora-packet-padio-breakouts>

Transceiver Moteino: <https://lowpowerlab.com/guide/moteino/transceivers/>

**RFM69**

Módulo basado en SX1231 con interfaz SPI

* +13 a +20 dBm hasta 100 mW Capacidad de salida de potencia (salida de potencia seleccionable en software)
* Drenaje de corriente de 50 mA (+13 dBm) a 150 mA (+20 dBm) para transmisiones, ~ 30 mA durante la escucha de radio activa.
* Las radios RFM69 tienen un alcance de aprox. Línea de visión de 500 metros con antenas unidireccionales sintonizadas. Dependiendo de las obstrucciones, la frecuencia, la antena y la potencia de salida, obtendrá rangos más bajos, especialmente si no tiene línea de visión.
* Crear redes multipunto con direcciones de nodo individuales
* Motor de paquete cifrado con AES-128

Guía completa del módulo de radio RFM69: <https://learn.sparkfun.com/tutorials/rfm69hcw-hookup-guide>

Librería Arduino RFM69: <https://github.com/LowPowerLab/RFM69>

Completa información RFM69: <http://www.hoperf.com/upload/rf/RFM69W-V1.3.pdf>

Explicación de librería RFM69 <https://lowpowerlab.com/2013/06/20/rfm69-library/>

**RFM95**

Módulo basado en LoRa® SX1276 con interfaz SPI

* Capacidad de salida de potencia de +5 a +20 dBm hasta 100 mW (salida de potencia seleccionable en software)
* ~ 100mA de pico durante la transmisión de + 20dBm, ~ 30mA durante la escucha activa de la radio.
* Las radios RFM9x tienen un rango de aprox. Línea de visión de 2 km con antenas unidireccionales sintonizadas. Dependiendo de las obstrucciones, la frecuencia, la antena y la potencia de salida, obtendrá rangos más bajos, especialmente si no tiene línea de visión.

Estos son radios de paquete LoRa de +20 dBm que tienen una modulación de radio especial que no es compatible con los RFM69 pero que puede ir mucho más lejos. Pueden ir fácilmente a la línea de vista de 2 km utilizando antenas de cable simples, o hasta 20 km con antenas direccionales y ajustes.

Completa información RFM95: <http://www.hoperf.com/upload/rf/RFM95_96_97_98W.pdf>

Librería: <http://www.airspayce.com/mikem/arduino/RadioHead/>

[SX127x Datasheet](https://cdn-shop.adafruit.com/product-files/3179/sx1276_77_78_79.pdf)– The RFM9X LoRa radio chip itself

Librería: <http://www.airspayce.com/mikem/arduino/RadioHead/>

**Módulos Moteino Usados**

Optamos LoRa porque da entre un 50% y 100% más de alcance.

LoRa support for Moteino: <https://lowpowerlab.com/guide/moteino/lora-support/>

Todo sobre moteino y como programarlo: <https://lowpowerlab.com/guide/moteino/>

Los moteino a usar con LoRa son los moteinoLR y mejor moteinoUSB-LoRa que ya tiene el interfaz USB:

* Moteino: <https://lowpowerlab.com/shop/product/99>
* Moteino USB: <https://lowpowerlab.com/shop/product/100>
* Transceiver RFM95: <https://lowpowerlab.com/shop/product/143>

