

Hardware de Control para Robots

Contenido

- Subsistemas del robot.
- Subsistema de alimentación.
- Subsistema de control de motores.
- Subsistema de comunicaciones.
- Subsistema de sensorización.
- Subsistema de control central.

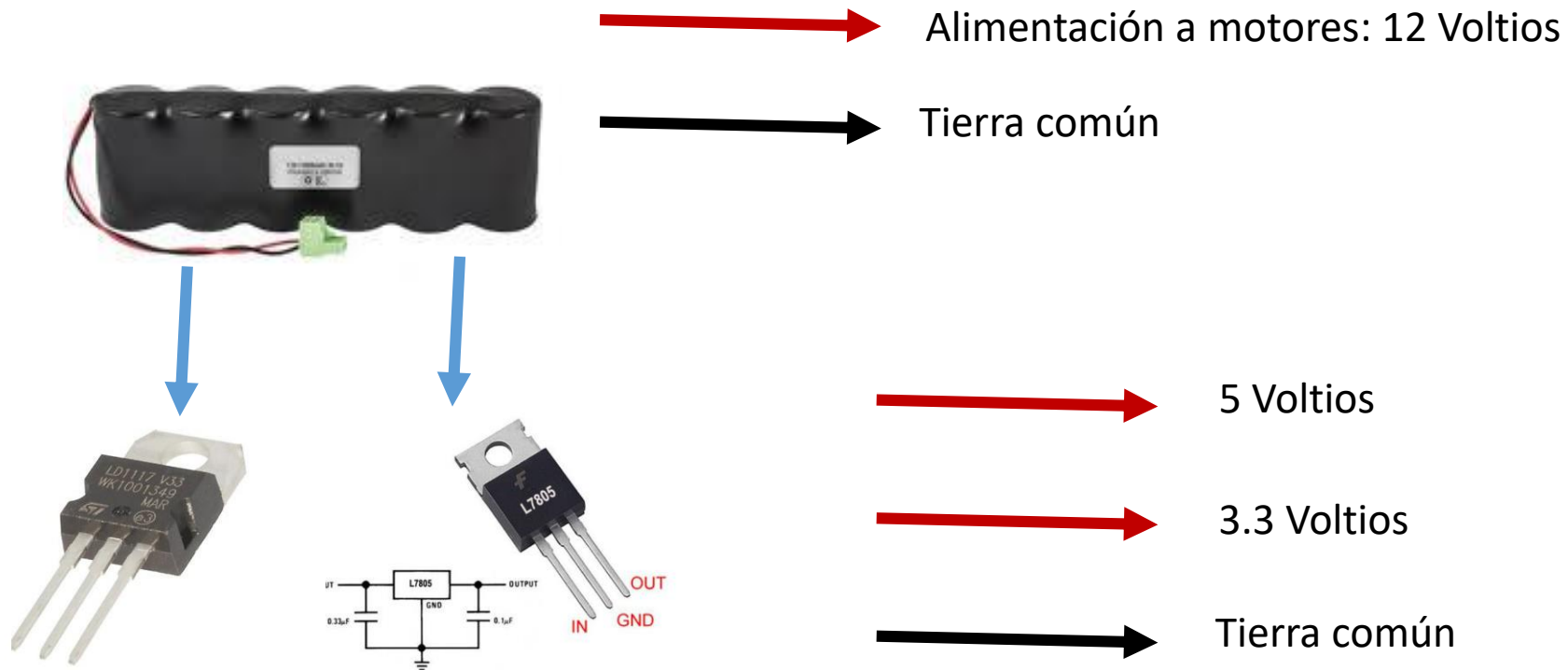
Subsistemas del Robot

- Robot → Subsistemas especializados en distintas tareas que le dotan de todas las características necesarias.
- Existe un subsistema fundamental → Alimentación eléctrica.
- Los subsistemas se alimentan del subsistema de alimentación eléctrica.
 - Debe tener la potencia necesaria para proveer la alimentación a cada uno de los subsistemas.
 - Comparten la tierra (tierra común).
- El subsistema de control central está formado por el/los microcontrolador/es correspondientes, que pueden cooperar entre sí.
- Cada subsistema → debe ser autónomo y enviar la información al control central.

Subsistema de Alimentación

- Formado por la/s batería/s principal/es y los adaptadores de tensión correspondientes.
- La potencia que entregue la batería debe ser superior a la potencia máxima que exijan los componentes del robot.
- Debe contemplarse un módulo de carga.
- Se deben seleccionar los adaptadores de tensión correspondientes a las tensiones de trabajo de los distintos subsistemas.

Subsistema de Alimentación



Subsistema de Alimentación

- Alimentación a motores principales.
 - Es recomendable un adaptador de tensión.
 - Los motores se conectan en configuración puente H.



Subsistema de Alimentación

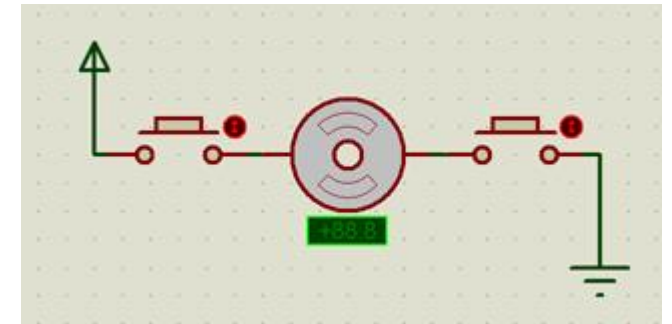
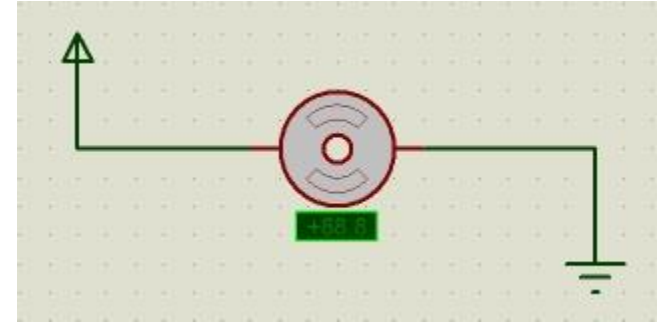
- Alimentación a motores principales.
 - Es recomendable un adaptador de tensión.
 - ¿Cómo se controlan los motores?.
 - ¿Directamente desde el microcontrolador de control central?.
 - Los motores se conectan en configuración puente H.

Subsistema de Control de Motores

- Para controlar la dirección en la que gira un motor de corriente continua, se necesita invertir la polaridad de la conexión eléctrica del motor en cuestión → de alguna manera hay que intercambiar los cables que alimentan el dispositivo.
- Si pensamos en una conexión normal como la de la siguiente imagen, vemos como cada terminal del motor recibe un polo distinto.
 - Con la conexión anterior conseguimos que el motor gire en un sentido cualquiera, si necesitamos que gire en el sentido contrario, basta con intercambiar las conexiones, pero esto solo es útil si no necesitamos conmutar rápidamente entre una dirección u otra.

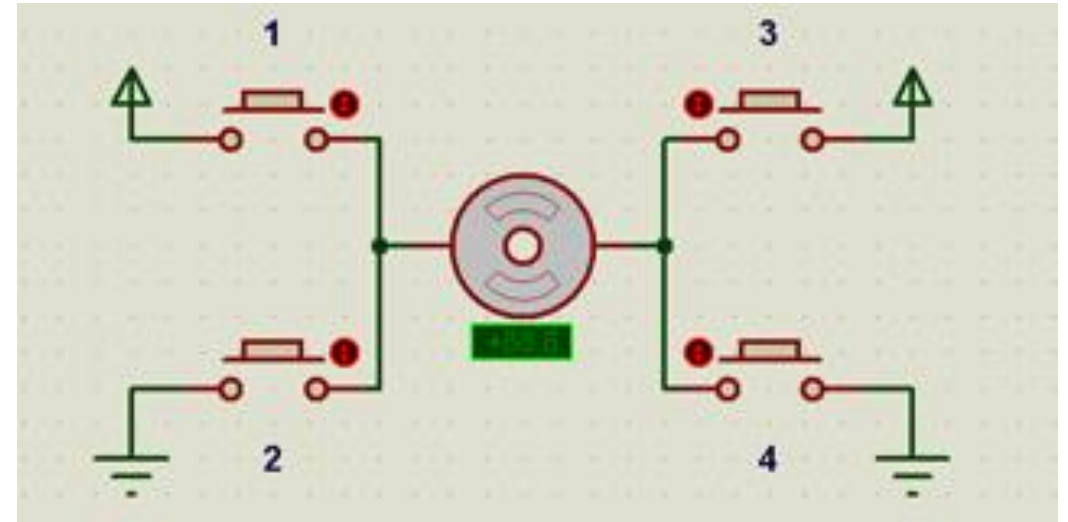
Subsistema de Control de Motores

- Si consideramos interruptores que controlen las conexiones, podríamos encontrarlos con la configuración donde un botón interrumpe el paso desde la alimentación a uno de los terminales del motor y otro botón, bloquea la conexión a tierra, si presionamos los dos botones al mismo tiempo el motor girará en un sentido.



Subsistema de Control de Motores

- Aprovechando este tipo de conexiones podríamos armar dos circuitos separados que compartan el mismo motor, así, si activamos un par de botones podemos hacer que gire en un sentido, y activando el otro par de botones, el motor gira en el sentido contrario.



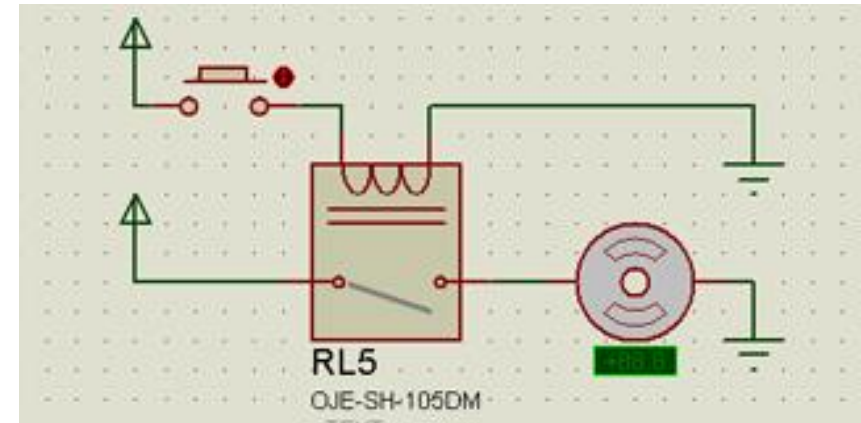
¿Cómo harías para mover el motor a derecha e izquierda?

Subsistema de Control de Motores

- A partir del concepto anterior, podemos implementar algún sistema de control remoto que nos permita accionar un motor junto al control de su dirección de giro.
- Sólo debemos mejorar el circuito para lograr por un lado, hacer actuar los pares de interruptores (1-4 y 2-3) a un control más simple y al mismo tiempo aislar el circuito del motor, del circuito de control, de esta manera podríamos manejar motores de un voltaje X, con un circuito que utiliza un voltaje Y.
- Incluso debemos tomar en cuenta que el motor al dejar de ser alimentado probablemente siga girando por inercia, en este caso se comportaría como un generador, entregando corriente al circuito que lo controla.
 - ¿Qué puede pasar?.

Subsistema de Control de Motores

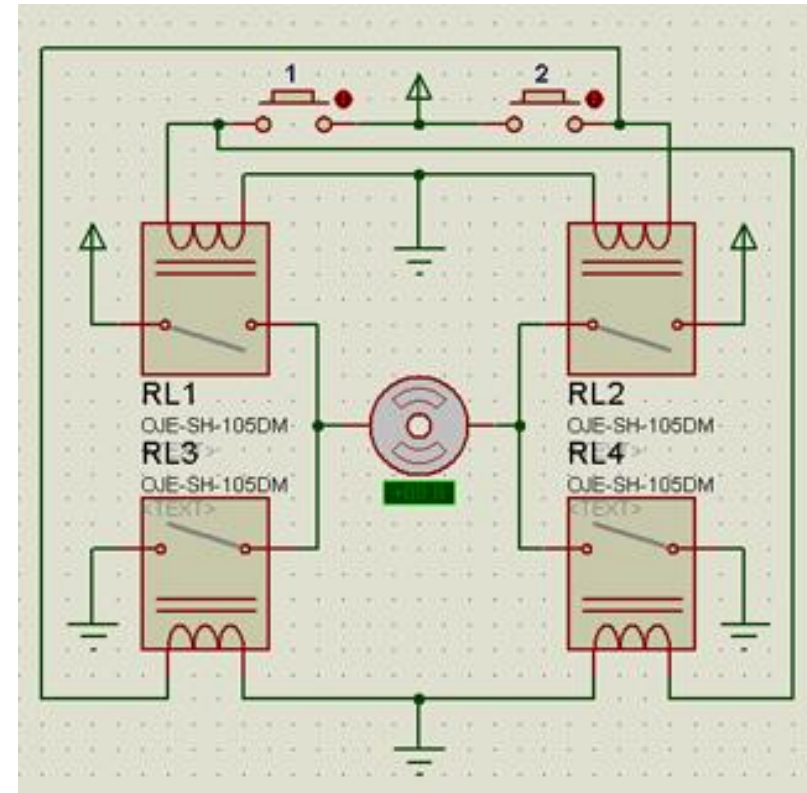
- Solución → El puente en H.
- El uso de relés para reemplazar los botones del circuito base podría ser una muy buena solución para aislar el circuito del motor del circuito de control.
- Los relés funcionan como botones mecánicos accionados eléctricamente, es decir, alimentamos un sistema eléctrico que a su vez acciona otro sistema eléctrico, aislado.
- Este mecanismo nos permite controlar voltajes altos, a partir de voltajes mucho menores.
- En la figura vemos como hay dos circuitos, por un lado el circuito base que activa el motor, se ve interrumpido por el relé. Por otro lado, está el circuito que controla el relé, este también está interrumpido, en este caso por un pulsador normalmente abierto. Al pulsar el interruptor, se cierra el circuito del relé y este se energiza, a su vez, el relé en funcionamiento, acciona el interruptor que controla el motor, cerrando este circuito, accionando el motor.



¿Cómo funciona el Relé?

Subsistema de Control de Motores

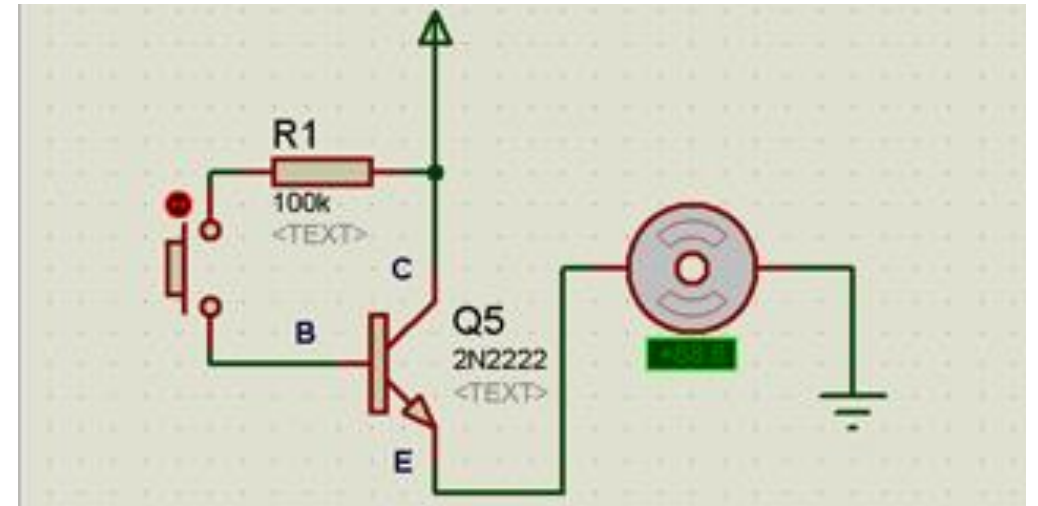
- Solución → El puente en H.
- Puente en H con los relés. Los cuatro botones han sido reemplazados por relés y la alimentación eléctrica de los relés está controlada por solo dos pulsadores, estos permiten accionar los relés por parejas, el pulsador 1 alimenta los relés RL1 y RL4 y el pulsador 2, RL2 y RL3, además podemos ver que el circuito que controla los relés puede ser completamente distinto (y aislado) del que alimenta el motor, incluso los circuitos que alimentan el motor para que gire en un sentido, puede ser distinto que alimenta el motor cuando gira en dirección contraria.



Puente en H con relés

Subsistema de Control de Motores

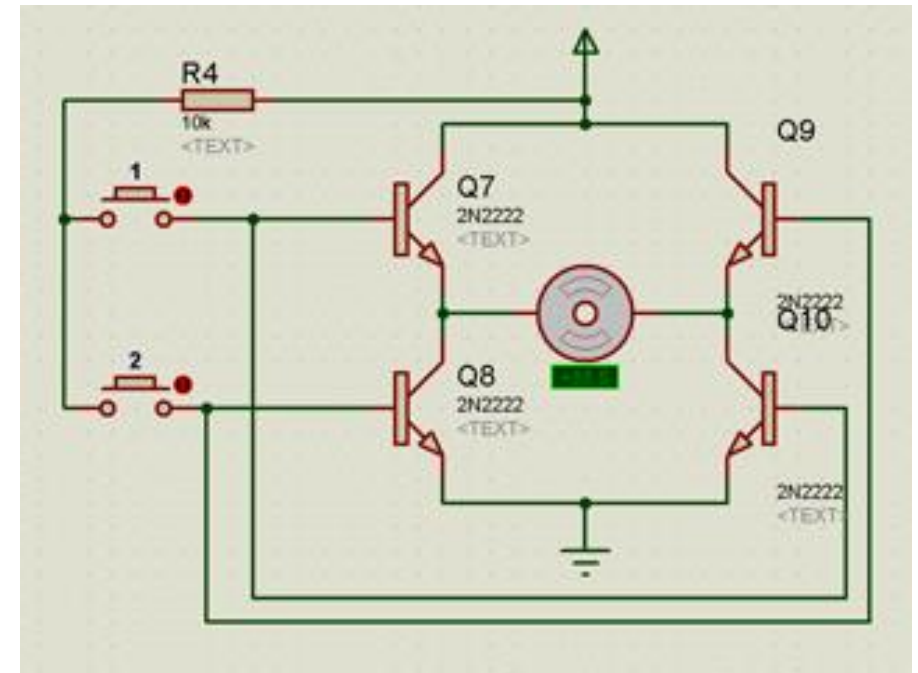
- Solución → El puente en H.
- Utilizando un transistor como interruptor, podemos replicar los esquemas anteriores. Recordemos que los transistores son componentes electrónicos, no electromecánicos como los relés vistos anteriormente, aquí el circuito es un poco más complejo.



Transistor para controlar un motor

Subsistema de Control de Motores

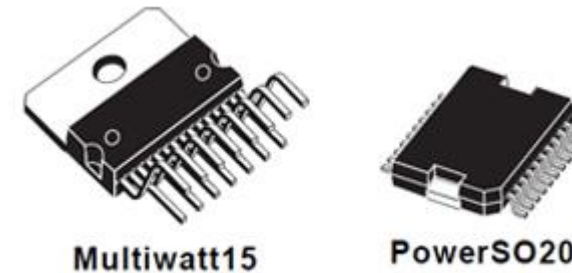
- Solución → El puente en H.
- El transistor actúa como una puerta desde un lado a otro, en este caso desde el colector al emisor, la base se comporta como el portero, si no recibe electricidad la puerta permanece cerrada, si recibe una señal, se abre.
- El voltaje que estimula la base es mucho menor que el que entra por el colector, por eso en este caso hay una resistencia a la base.



Puente en H con transistores

Subsistema de Control de Motores

- Solución → El puente en H.
- El circuito de un Puente H es muy utilizado, en robótica debe ser uno de los imprescindibles, es por eso que, además de una infinidad de circuitos desarrollados a partir de componentes sueltos, podemos encontrarlos como circuitos integrados.
- Hay una variedad suficiente como para desarrollar pequeños proyectos sin invertir en los gastos de un desarrollo más grande, aunque en general no son componentes baratos.
- El modelo L298 es el más utilizado en robótica para el control de motores DC.



L298

Subsistema de Control de Motores



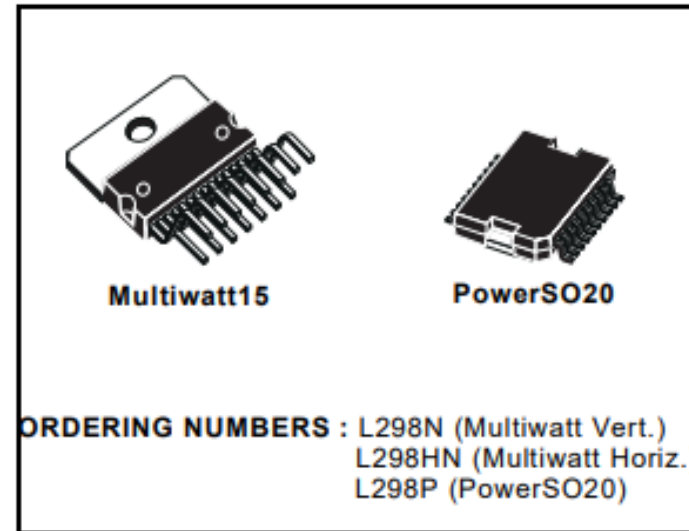
L298

DUAL FULL-BRIDGE DRIVER

- OPERATING SUPPLY VOLTAGE UP TO 46 V
- TOTAL DC CURRENT UP TO 4 A
- LOW SATURATION VOLTAGE
- OVERTEMPERATURE PROTECTION
- LOGICAL "0" INPUT VOLTAGE UP TO 1.5 V (HIGH NOISE IMMUNITY)

DESCRIPTION

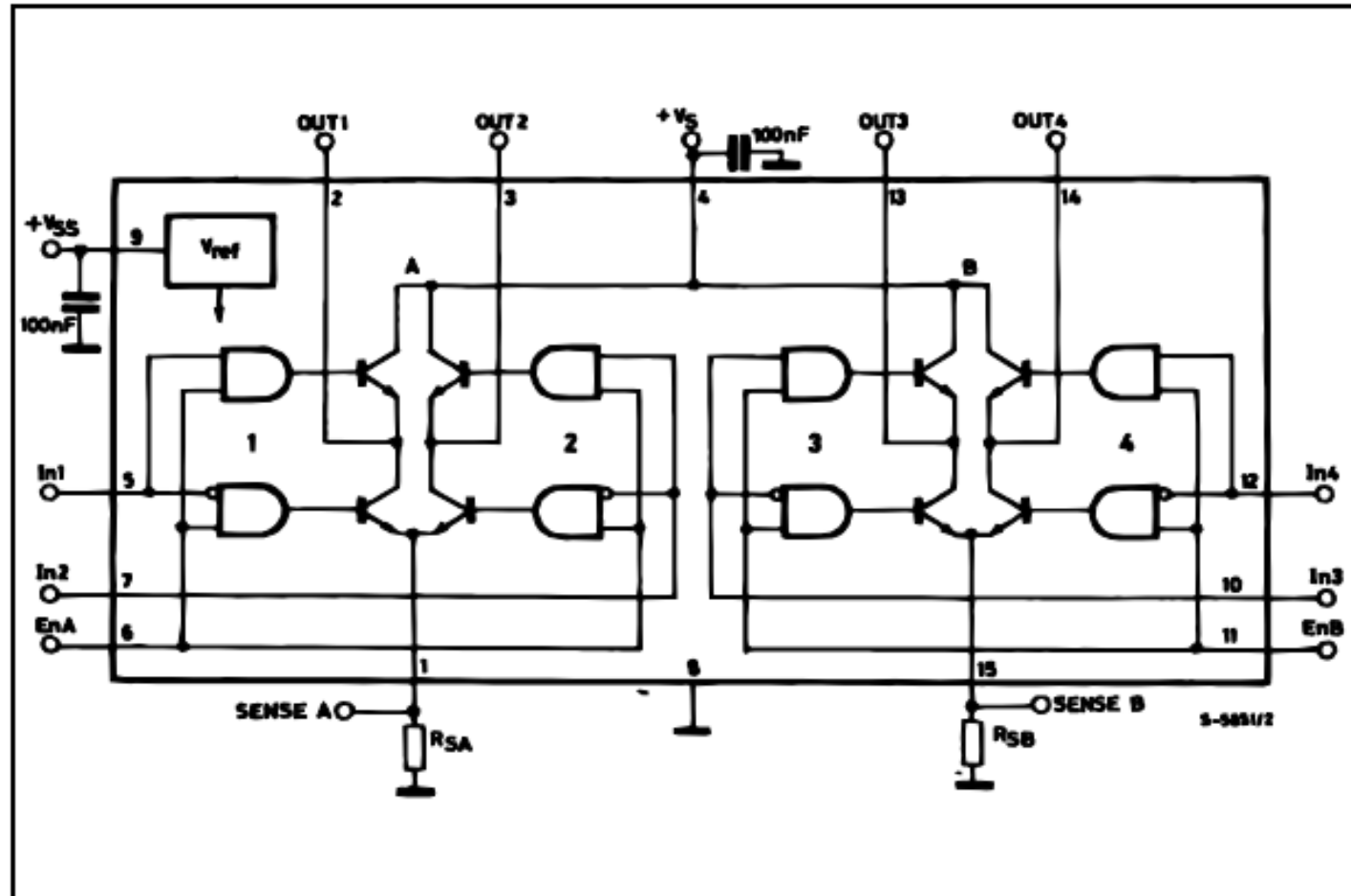
The L298 is an integrated monolithic circuit in a 15-lead Multiwatt and PowerSO20 packages. It is a high voltage, high current dual full-bridge driver designed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepping motors. Two enable inputs are provided to enable or disable the device independently of the input signals. The emitters of the lower transistors of each bridge are connected together and the corresponding external terminal can be used for the con-



nection of an external sensing resistor. An additional supply input is provided so that the logic works at a lower voltage.

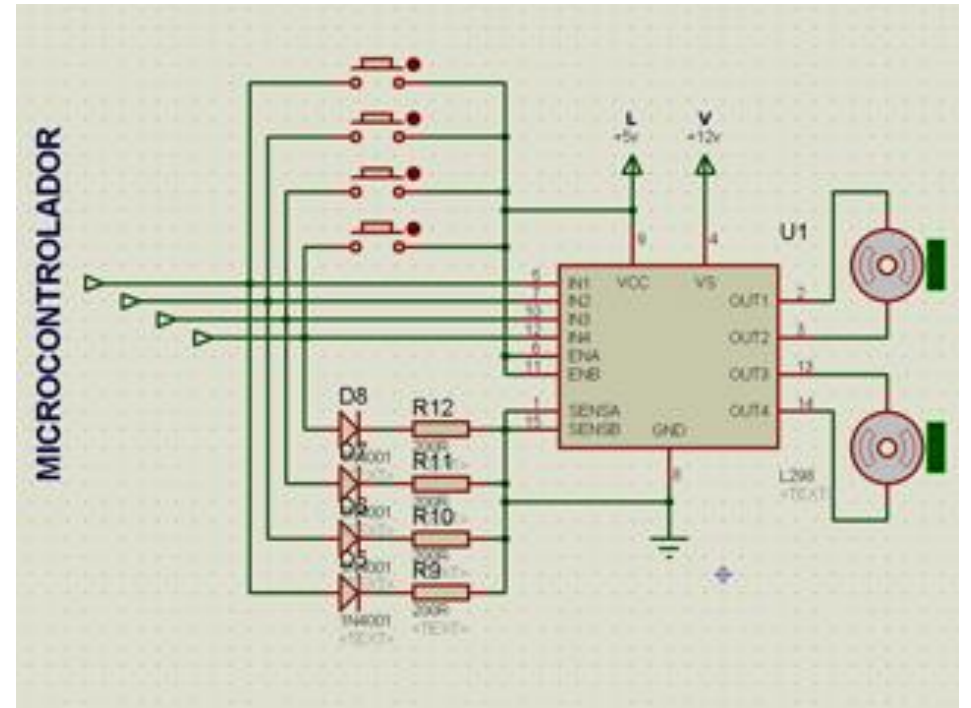
Subsistema de Control de Motores

BLOCK DIAGRAM



Subsistema de Control de Motores

- Solución → El puente en H.
- El L298, en realidad, son cuatro circuitos independientes, que funcionan como medio Puente H cada uno, de esta manera podemos utilizarlo para controlar dos motores independientemente solo es cosa de orden al armar la circuitería.

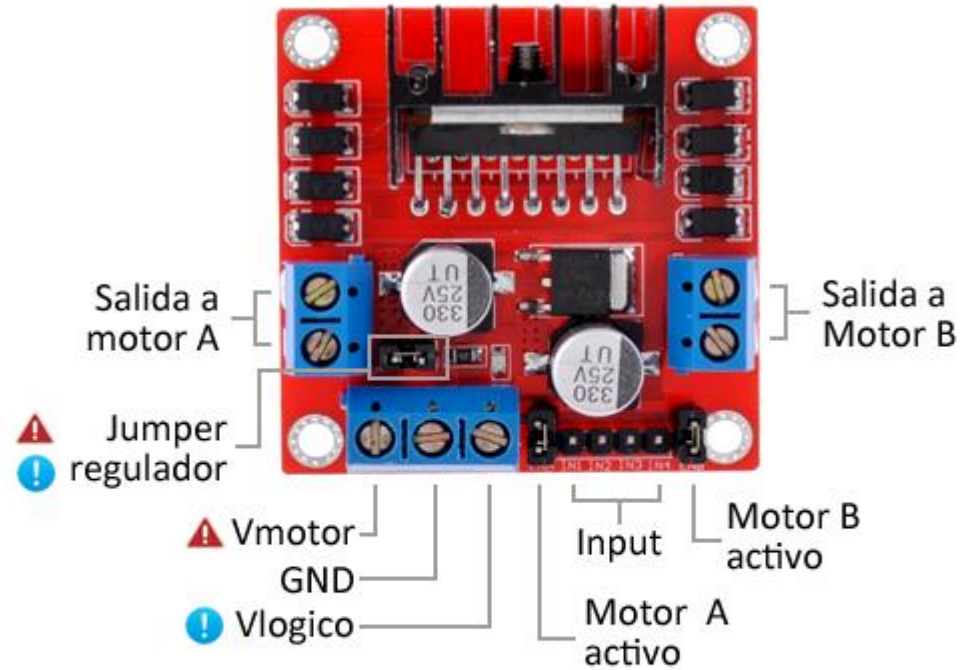


L298

Subsistema de Control de Motores

- El L298N es un controlador (driver) de motores, que permite encender y controlar **dos motores de corriente continua desde Arduino**, variando tanto la dirección como la velocidad de giro.
- Arduino, y en general todos los autómatas o microcontroladores, no disponen de potencia suficiente para mover actuadores. De hecho, la función de un procesador no debe ser ejecutar acciones si no mandar ejecutar acciones a drivers que realicen el “trabajo pesado”.
- La corriente máxima que el L298N puede suministrar a los motores es, en teoría, **2A por salida (hasta 3A de pico) y una tensión de alimentación de 3V a 35V**.
- El L298N incorpora protecciones contra efectos que pueden producirse al manejar motores de corriente continua. Dispone de protecciones contra sobre intensidad, sobre temperatura, y diodos de protección contra corrientes inducidas (flyback).

Subsistema de Control de Motores



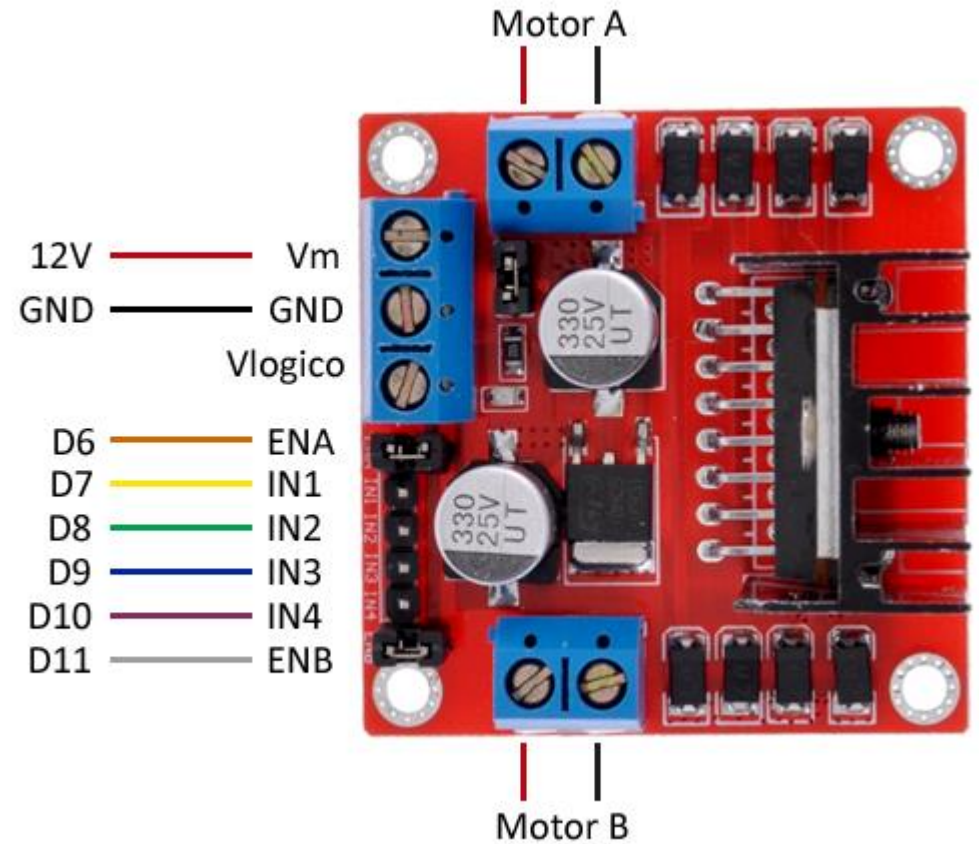
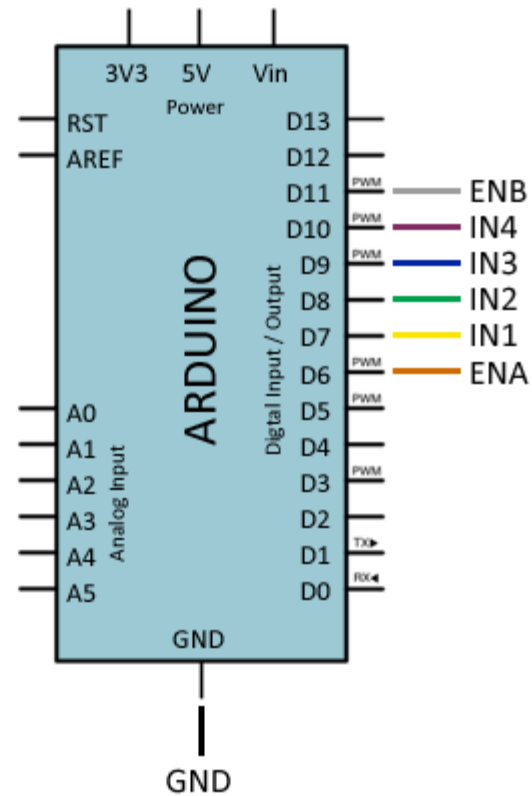
Con Jumper cerrado: Vmotor entre 6V y 12V
Con Jumper abierto: Vmotor entre 12V y 35V



Con Jumper cerrado: Vlogico salida de 5V
Con Jumper abierto: Vlogico entrada de 5V



Subsistema de Control de Motores



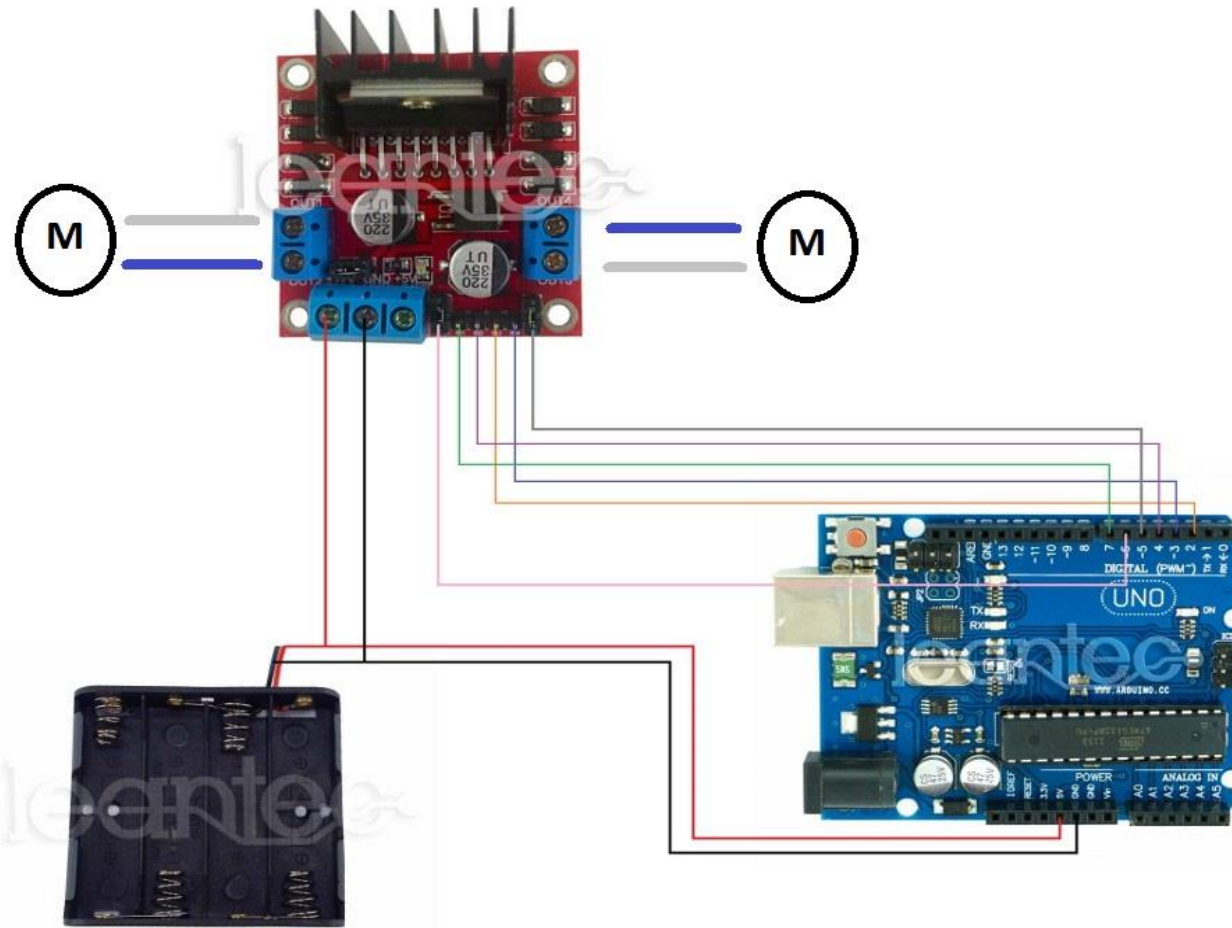
¿Qué pasa si aplicamos PWM?

Subsistema de Control de Motores

- Ejemplo de código.
 - El siguiente código mueve hacia adelante el motor de la salida A al 80% de velocidad.

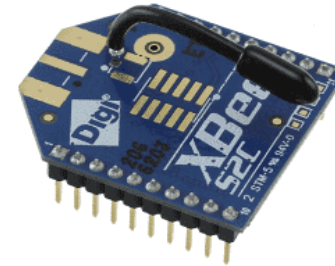
```
1  const int pinENA = 6;
2  const int pinIN1 = 7;
3  const int pinIN2 = 8;
4
5  const int speed = 200;    //velocidad de giro 80% (200/255)
6
7  void setup()
8  {
9      pinMode(pinIN1, OUTPUT);
10     pinMode(pinIN2, OUTPUT);
11     pinMode(pinENA, OUTPUT);
12 }
13
14 void loop()
15 {
16     digitalWrite(pinIN1, HIGH);
17     digitalWrite(pinIN2, LOW);
18     analogWrite(pinENA, speed);
19     delay(1000);
20 }
```


Subsistema de Control de Motores



Subsistema de Comunicaciones

- Se suelen utilizar comunicaciones por radio para el control o monitorización de un robot.
- Tipos de comunicaciones inalámbricas.
 - WiFi.
 - RF433.
 - Bluetooth.
 - ZigBee.
 - LORA.
 - SigFox.



Subsistema de Comunicaciones

- RF433.
 - Los módulos de radio frecuencia RF 433MHz son transmisores/receptores inalámbricos que podemos emplear como forma de comunicación entre procesadores/microcontroladores.
 - Este tipo de módulos emisor (FS1000A) y el receptor (XY-MK-5V) se han hecho populares como medio de comunicación, principalmente, por su bajo coste.
 - La frecuencia de operación es de 433MHz, aunque también existen módulos similares a 315MHz. Ambas frecuencias pertenecen a bandas libres, por lo que su uso es gratuito.
 - El alcance depende del voltaje con el que alimentemos el módulo y la antena que usemos. A 5V y con la antena del módulo, el alcance difícilmente excederá de los 2 metros. Alimentando a 12V y con una antena de cobre de 16.5cm el rango en exteriores puede alcanzar 300 metros.

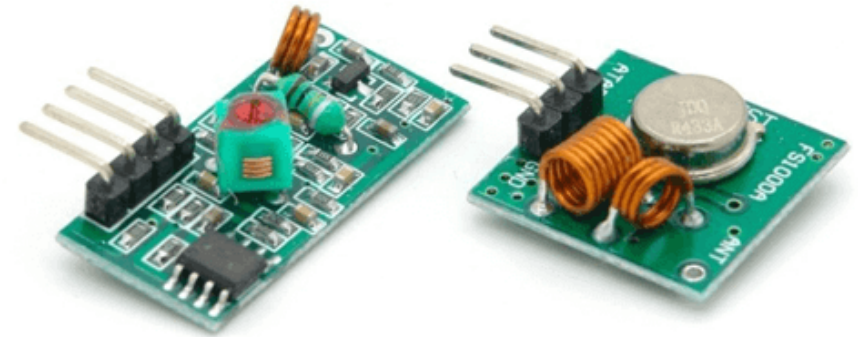
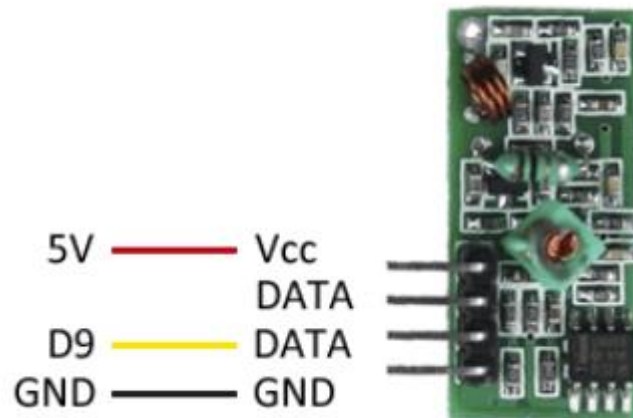
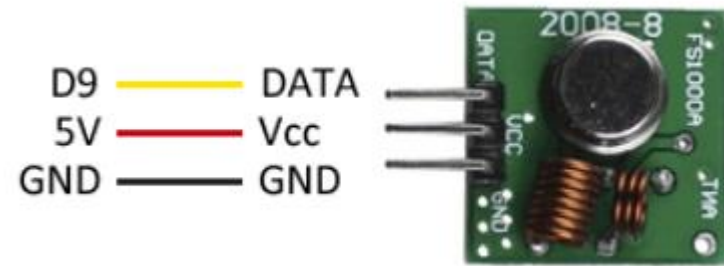


Subsistema de Comunicaciones

- RF433.
 - La comunicación es simplex (canal único y unidireccional) y tienen baja velocidad de transmisión (típicamente 2400bps).
 - Se realiza, básicamente, por modulación ASK (amplitude shift keying).
 - No disponen de filtro ni ID por hardware, por lo que si queremos una comunicación robusta tendremos que implementarlo por software.
 - Este tipo de comunicación RF 433MHz es muy frecuente en proyectos caseros de electrónica y robótica, debido a su bajo precio y medio-largo alcance en condiciones adecuadas, aunque tenemos que tener en cuenta que la robustez y calidad de la comunicación puede ser limitada.
 - Ejemplos de uso son recepción remota de sensores como temperatura presión, aplicaciones de domótica y edificios inteligentes, activación remota de dispositivos como iluminación, alarmas, o control y monitorización de robots y vehículos.

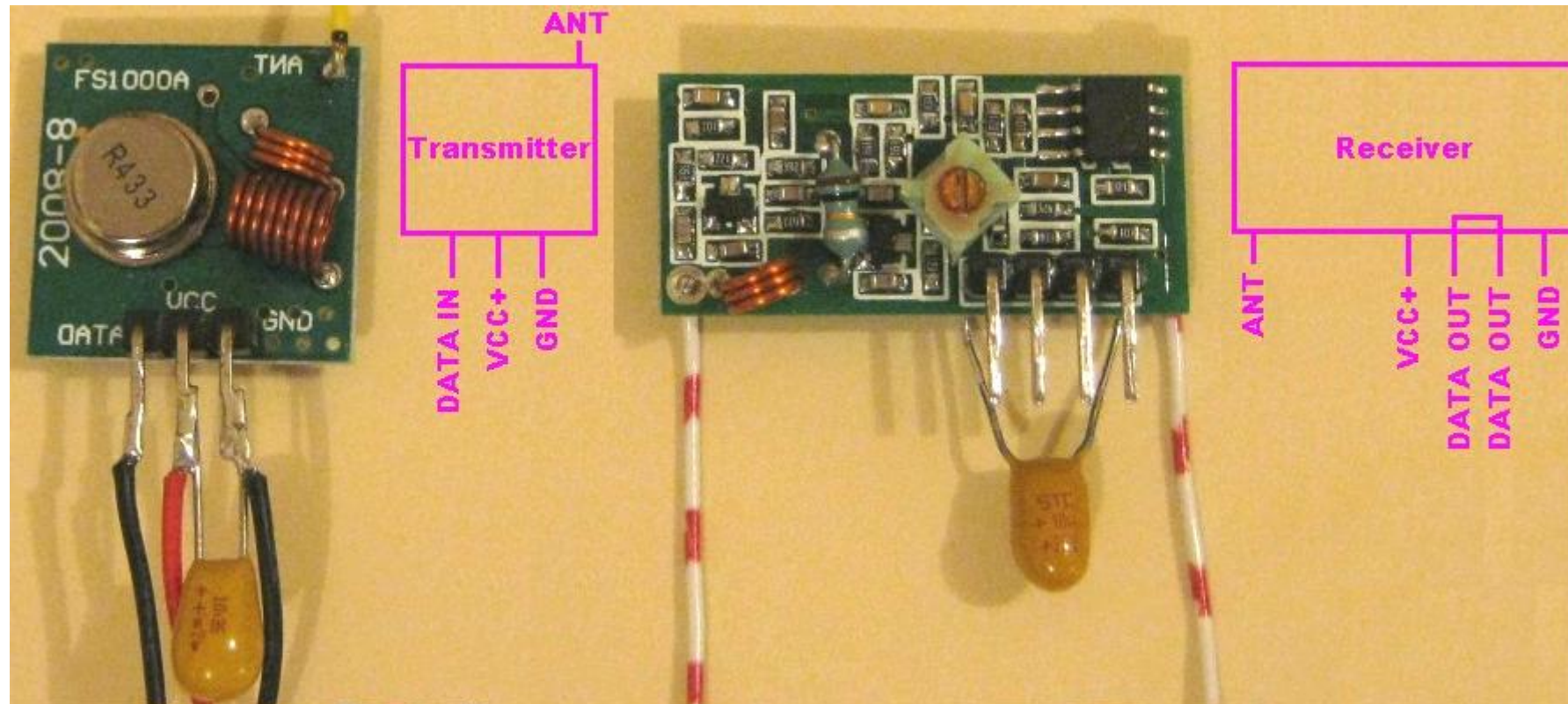


Subsistema de Comunicaciones



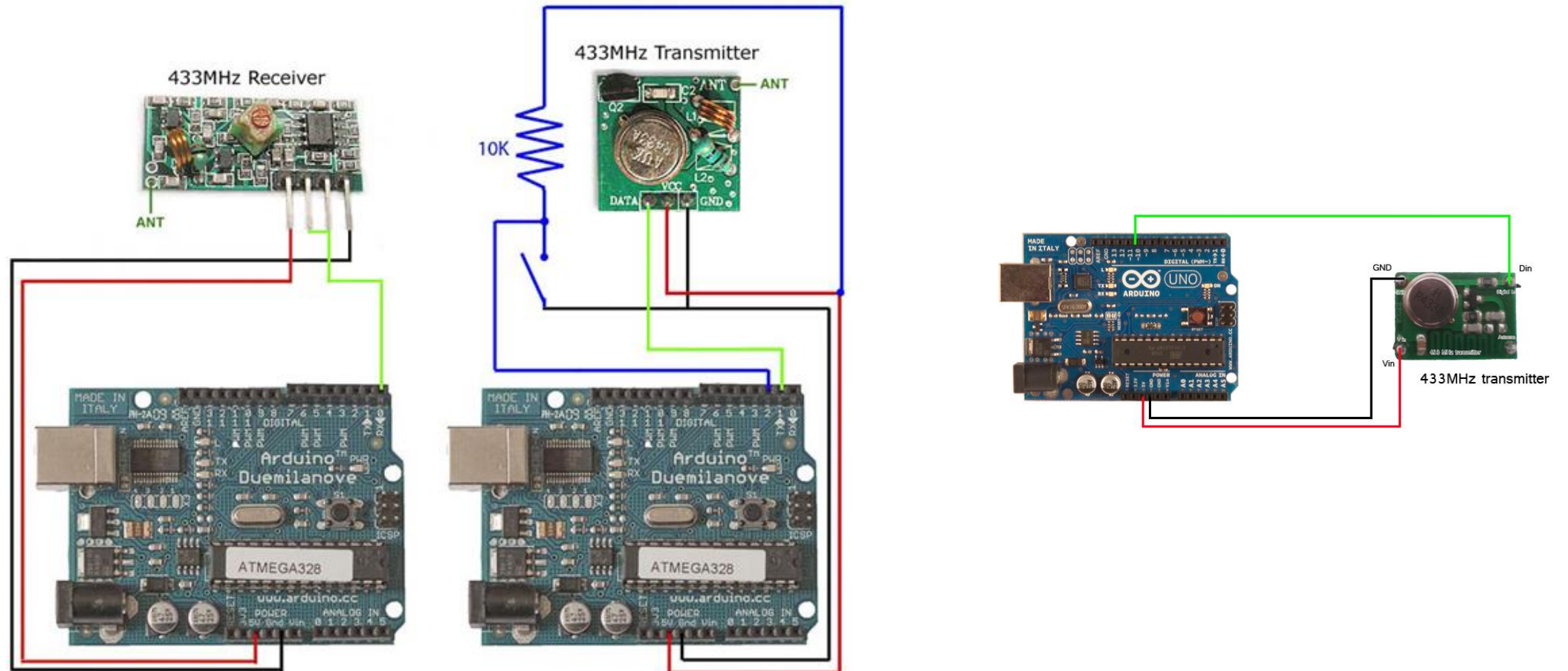
Subsistema de Comunicaciones

- RF433.



Subsistema de Comunicaciones

- RF433.



Subsistema de Comunicaciones

- Bluetooth (BLE).
 - Bluetooth Low Energy (BLE), a veces conocido como “Bluetooth Smart”, se introdujo como parte de la especificación de Bluetooth 4.0.
 - Existen un montón de protocolos wireless para uso en IOT, pero lo que hace que BLE sea tan interesante es que es el más sencillo para implementar la comunicación entre pequeños dispositivos y una aplicación en cualquier plataforma móvil actual (iOS, Android, Windows phones, etc.).
 - En el caso de los dispositivos Apple, es el único método que permite la interacción de periféricos con aplicaciones, sin necesidad de certificaciones MFI y otros requisitos legales que exige iOS.



Subsistema de Comunicaciones

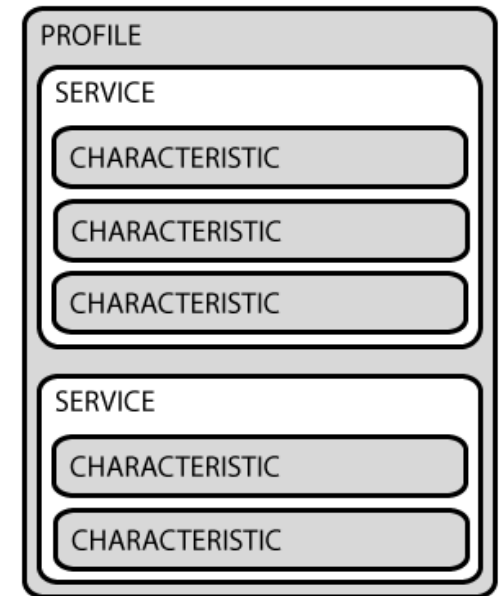
- Bluetooth (BLE).
 - Qué es el GAP?. Es el acrónimo para el **Generic Access Profile**, y se encarga de controlar las conexiones y los anuncios en BLE. GAP es lo que permite que tu dispositivo sea público hacia el exterior y determina como dos dispositivos pueden (o no) interactuar entre ellos.
 - GATT. es el acrónimo de Generic Attribute Profile, y define la manera en que dos dispositivos BLE pueden comunicarse usando los Servicios y Características. La comunicación se realiza mediante un protocolo conocido como ATT, que se usa para almacenar los servicios, características y datos relacionados en una tabla usando identificadores de 16-bit para cada entrada en la tabla.
 - GATT entra en juego una vez se ha establecido una conexión dedicada entre dos dispositivos, lo que significa que ya hemos pasado previamente por el GAP.

Subsistema de Comunicaciones

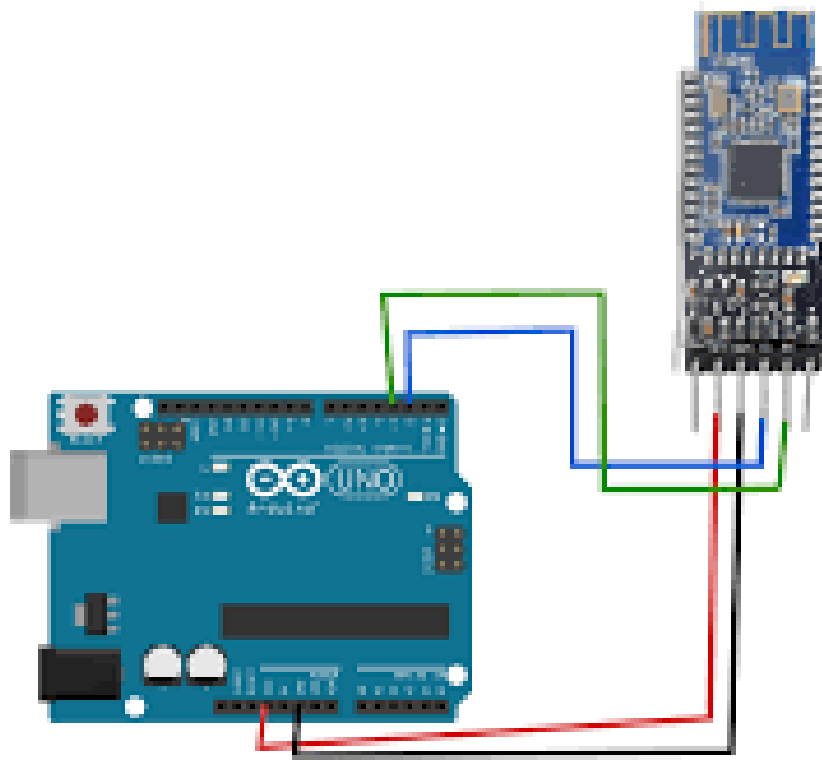
- Bluetooth (BLE).
 - Lo más importante a tener en cuenta con el GATT y las conexiones, es que las conexiones son exclusivas. **Un periférico BLE sólo puede ser conectado a un dispositivo central (un teléfono móvil, etc) a la vez.** Tan pronto como un periférico se conecta a un dispositivo central, dejará de anunciarse y otros dispositivos ya no podrán verlo o conectarse a él, hasta que se finalice la conexión existente.
 - El GATT es una relación servidor / cliente.
 - **El periférico se conoce como el servidor GATT**, que contiene los datos de búsqueda ATT y las definiciones de servicio y características, **y el cliente GATT (el teléfono / tableta)**, que envía solicitudes a este servidor.

Subsistema de Comunicaciones

- Bluetooth (BLE).
 - **Todas las transacciones son iniciadas por el dispositivo maestro**, el GATT Client (central) , que recibe la respuesta del dispositivo esclavo, el GATT Server (periférico).
 - Las transacciones GATT en BLE se basan en objetos anidados de alto nivel denominados Perfiles, Servicios y Características.
 - Se puede encontrar una lista completa de los servicios BLE adoptados oficialmente en la página del Portal de desarrolladores de Bluetooth. Por ejemplo, si observamos el Servicio de Frecuencia Cardíaca, podemos ver que este servicio oficialmente adoptado tiene un UUID de 16 bits de 0x180D, y contiene hasta 3 características, aunque sólo es obligatorio la primera: Medición de la frecuencia cardíaca, Sensor corporal Ubicación y Punto de Control de la Frecuencia Cardíaca.



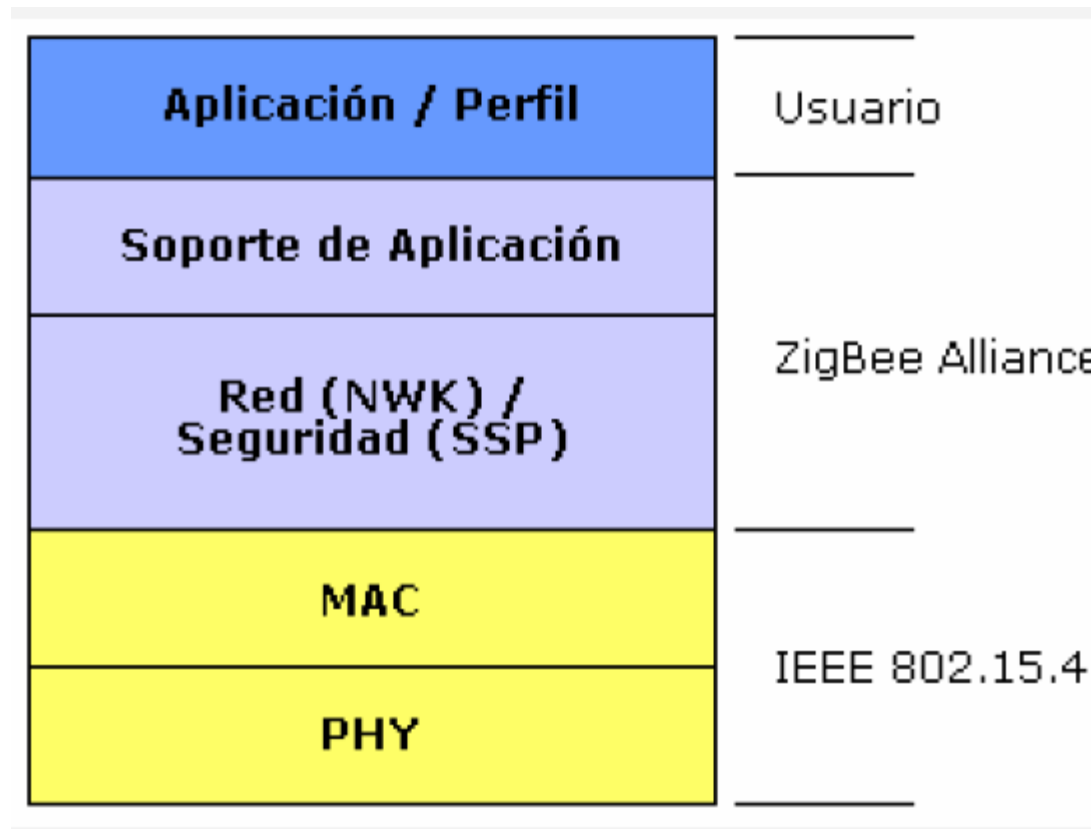
Subsistema de Comunicaciones



Subsistema de Comunicaciones

- ZigBee.
 - **Se trata de un sistema de comunicación inalámbrica** centrado en la comunicación entre dispositivos con una baja tasa de datos con el fin de tener el menor consumo energético posible. Una tecnología centrada en la domótica.
 - La tecnología de comunicación inalámbrica ZigBee utiliza la **banda ISM** y por lo general, **adopta la banda 2.4GHz** para comunicarse con el resto de dispositivos ya que esta se adopta en todo el mundo.
 - Una de las principales ventajas de ZigBee es lo **sencillo y el bajo coste que supone para la empresa** producir dispositivos con esta tecnología de comunicación.
 - Mucho más sencillo que Bluetooth ya que requiere un **10% del hardware total necesario para producir un dispositivo con bluetooth.**

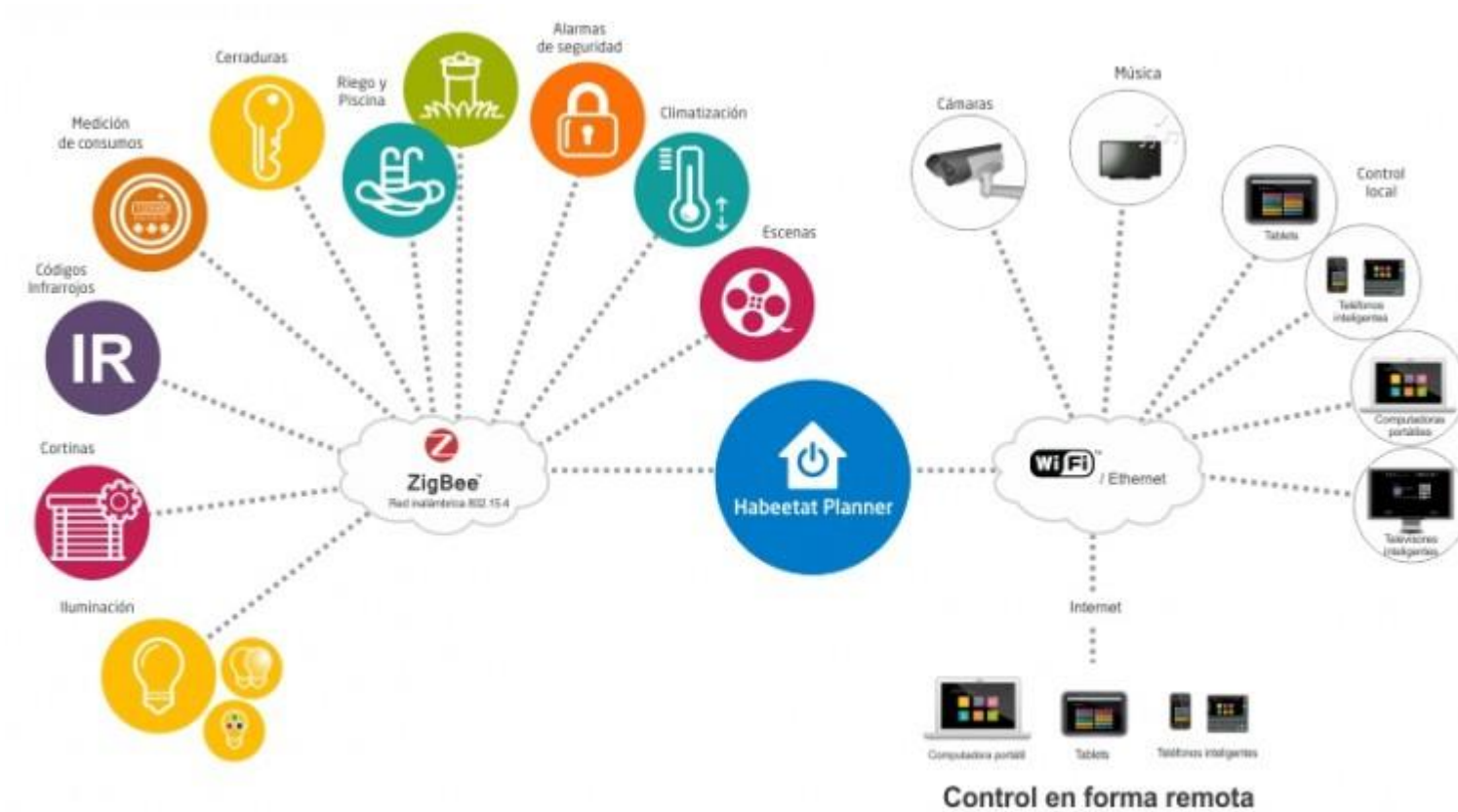
Subsistema de Comunicaciones



Subsistema de Comunicaciones

- ZigBee.
 - Una red formada por dispositivos ZigBee puede tener distintas topología: **Estrella, árbol y malla**, siendo la más usada la organización en malla. Así pues, un nodo ZigBee puede estar conectado a su vez a otros más de la misma red. De este modo, se asegura la comunicación entre todos los nodos porque siempre habrá un camino para seguir en caso de caída de uno. Claro está, el nodo coordinador es el que dirige el paso de mensajes entre cada nodo de la malla.

Subsistema de Comunicaciones



Subsistema de Comunicaciones

- ZigBee.
- Tres categorías de nodos:
 - El primero y más importante ya que ha de existir obligatoriamente en una red. El **Coordinador ZigBee** es el nodo más completo y se encarga de controlar toda la red y los caminos para su comunicación.
 - Por debajo tenemos el **Router ZigBee** que interconecta los nodos para poder ejecutar código del usuario, es decir, ofrece un nivel de aplicación dentro de la torre de protocolos.
 - Por último, el **dispositivo final ZigBee** sólo recibe información y se comunica únicamente con el nodo padre. La ventaja de este dispositivo es que puede permanecer dormido y 'despertarse' en ciertos momentos para alargar la duración de batería.

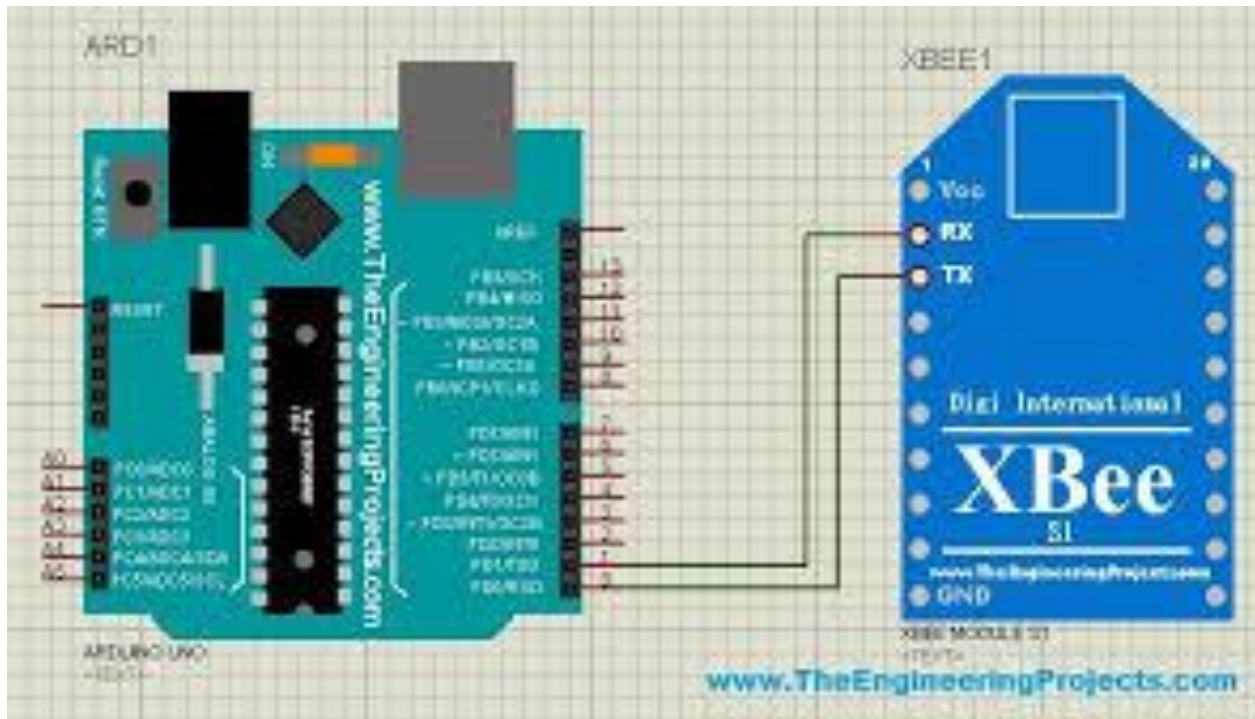
Subsistema de Comunicaciones

- ZigBee.
 - Por tanto, creando un nodo central con conexión a internet y con suministro energético constante se podría conectar toda una casa.
 - Un futuro que cada vez se ve más cerca, toda nuestra casa conectada y gestionada desde nuestro smarphone Android.



Subsistema de Comunicaciones

- ZigBee.



Subsistema de Comunicaciones

- LORA.
 - LoRa es el tipo de modulación en radiofrecuencia patentado por Semtech y que entre sus principales ventajas se encuentra:
 - Alta tolerancia a las interferencias
 - Alta sensibilidad para recibir datos
 - Basado en modulación chirp
 - Bajo Consumo (hasta 10 años con una batería*)
 - Largo alcance 10 a 20km
 - Baja transferencia de datos (hasta 255 bytes)
 - Conexión punto a punto
 - Frecuencias de trabajo: 915Mhz América, 868 Europa, 433 Asia

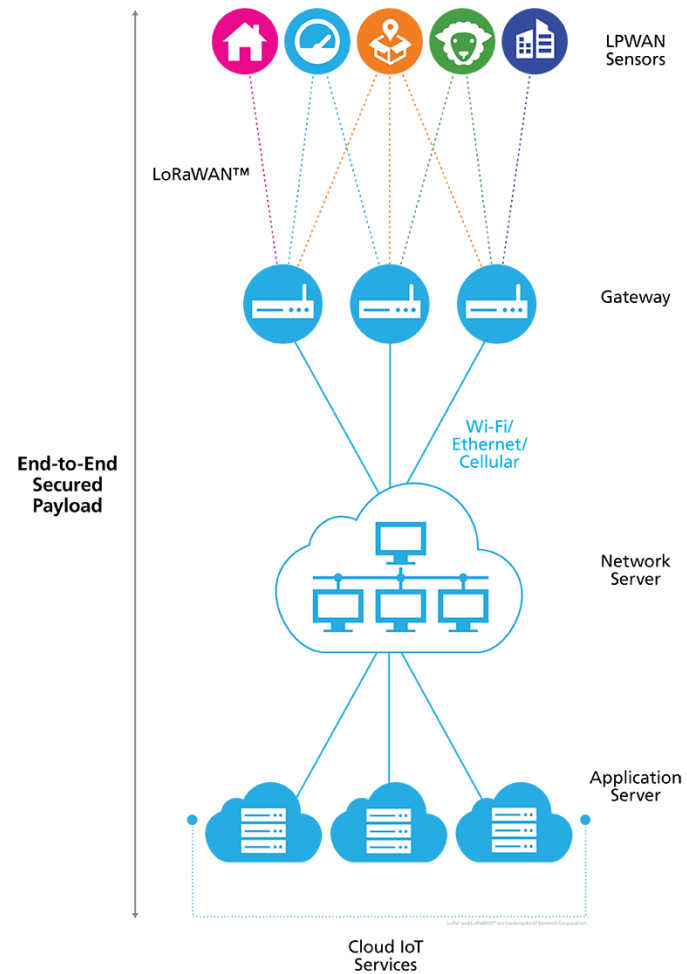
Subsistema de Comunicaciones

- LORA.
 - Todo esto hace a la tecnología ideal para conexiones a grandes distancias y para redes de IoT que se pueden utilizar en ciudades inteligentes, lugares con poca cobertura celular o redes privadas de sensores o actuadores, por eso es que nace LoRaWAN.
 - LoRaWAN es un protocolo de red que usa la tecnología LoRa para comunicar y administrar dispositivos LoRa, se compone de dos partes principalmente: gateways y nodos, los primeros son los encargados de recibir y enviar información a los nodos y los segundos, son los dispositivos finales que envían y reciben información hacia el gateway.

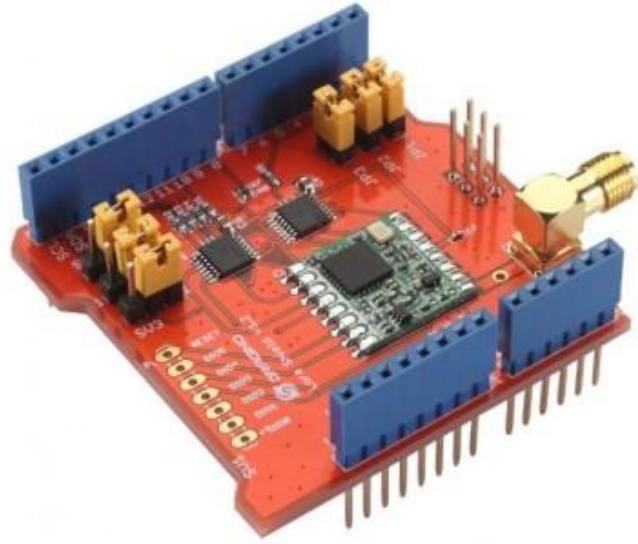
Subsistema de Comunicaciones

- LORA.
 - Conexiones punto a punto (P2P) o máquina a máquina.
 - Redes de sensores en ciudades, campo o industria.
 - Redes IoT donde NO se requiere transferir voz o video.
 - Tracking de vehículos, animales o personas.
 - Redes privadas que no requieren conectarse a servicios en la nube o donde no hay cobertura celular.
 - Gracias a su largo alcance y bajo consumo esto la hace una tecnología barata y fácil de implementar.

Subsistema de Comunicaciones



Subsistema de Comunicaciones



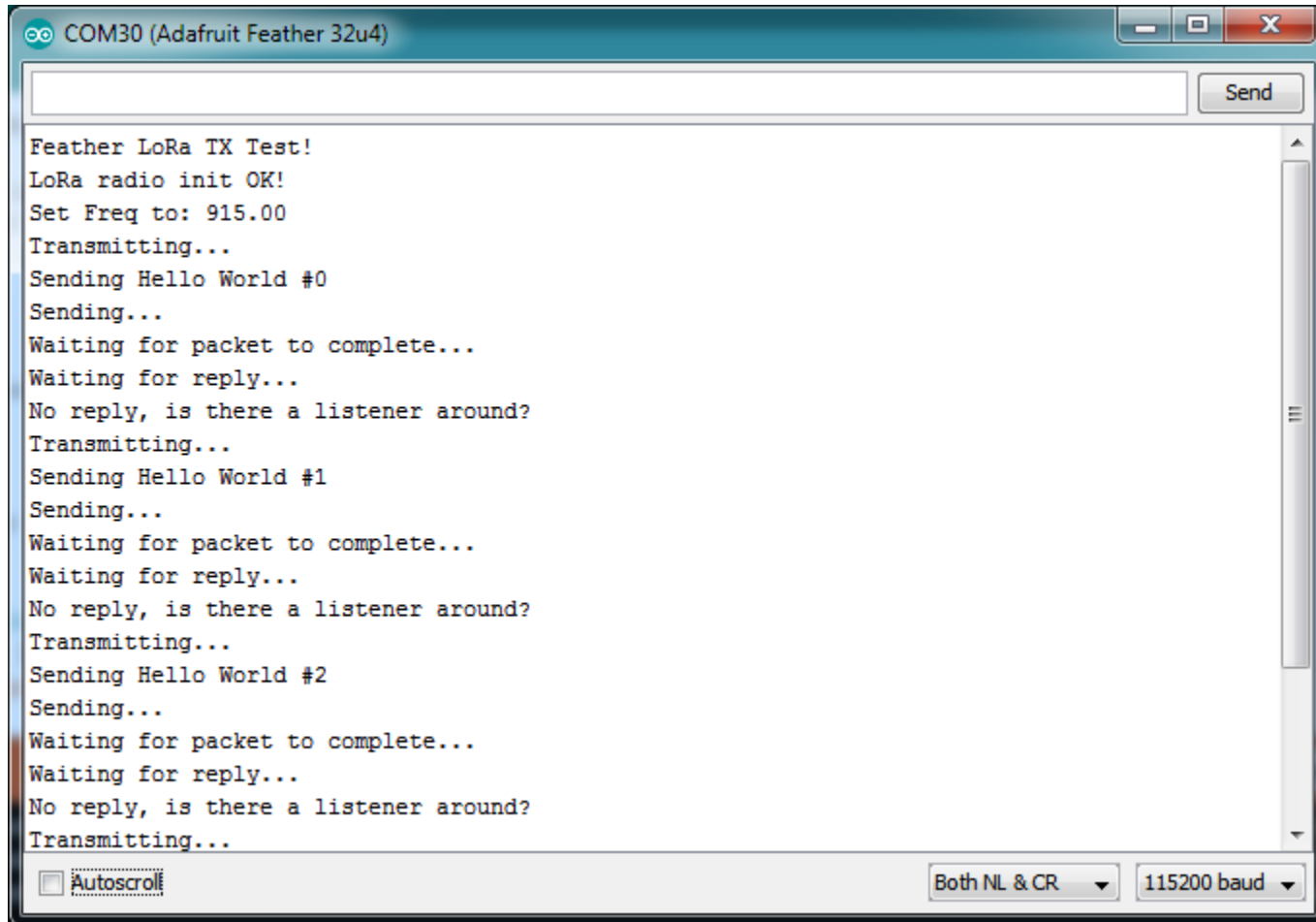
Emisor LoRa de larga distancia para Arduino
basado en el RFM98W a 433 MHz

- Compatible con placas Arduino de 3.3v y 5v
- Frecuencia de funcionamiento: 433 MHz
- Bajo consumo
- Ancho de banda programable de hasta 300 kbps.
- Buena inmunidad a las interferencias
- Recepción de bajo consumo: 10.3 mA, 200 nA register retention.
- Modulación: FSK, GFSK, MSK, GMSK, LoRa™ y OOK
- RRSI dinámico de 127 dB
- Paquetes de hasta 256 bytes con CRC.
- Sensor de temperatura integrado con detección de batería baja

Subsistema de Comunicaciones

- LORA.
 - Librería ARDUINO: RadioHead. Puede ser portada a otros microcontroladores con unos mínimos cambios.
 - <http://www.airspayce.com/mikem/arduino/RadioHead/>
 - [Código Transmisor.](#)
 - [Código Receptor.](#)

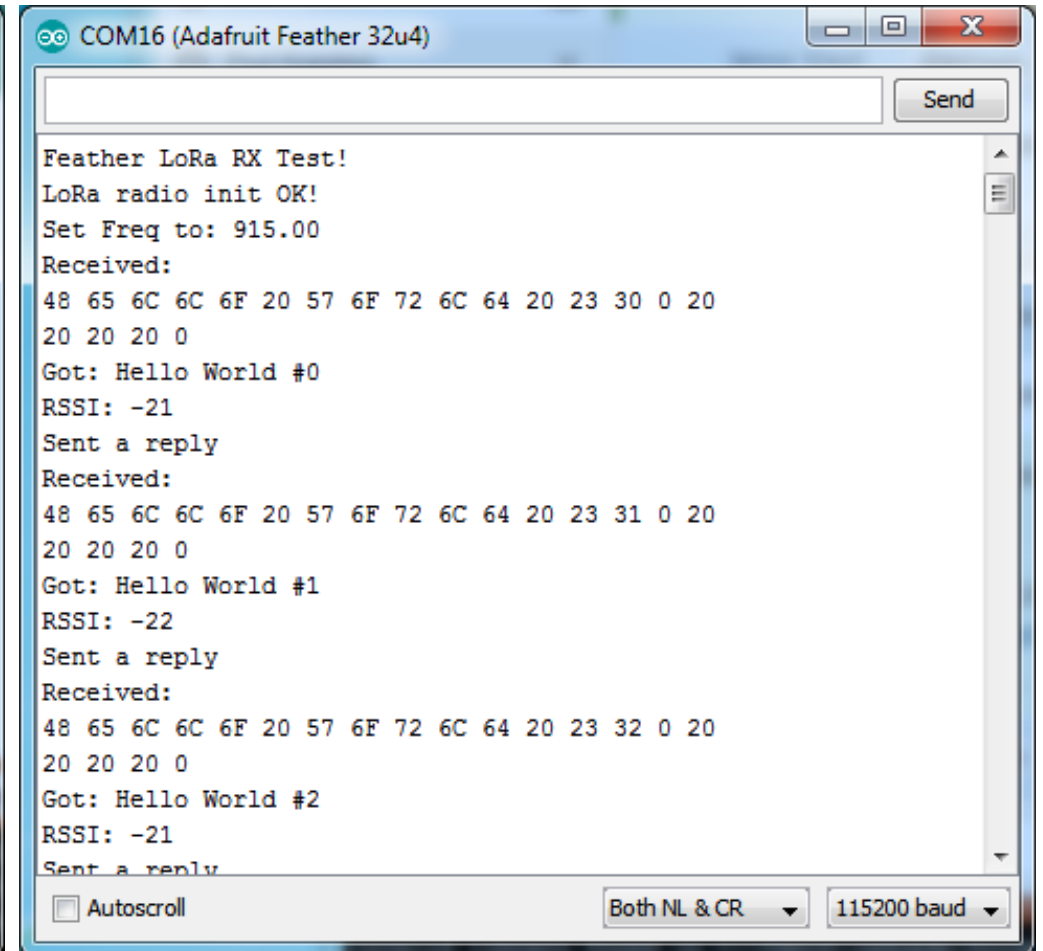
Subsistema de Comunicaciones



COM30 (Adafruit Feather 32u4)

```
Feather LoRa TX Test!  
LoRa radio init OK!  
Set Freq to: 915.00  
Transmitting...  
Sending Hello World #0  
Sending...  
Waiting for packet to complete...  
Waiting for reply...  
No reply, is there a listener around?  
Transmitting...  
Sending Hello World #1  
Sending...  
Waiting for packet to complete...  
Waiting for reply...  
No reply, is there a listener around?  
Transmitting...  
Sending Hello World #2  
Sending...  
Waiting for packet to complete...  
Waiting for reply...  
No reply, is there a listener around?  
Transmitting...
```

☐ Autoscroll Both NL & CR 115200 baud



COM16 (Adafruit Feather 32u4)

```
Feather LoRa RX Test!  
LoRa radio init OK!  
Set Freq to: 915.00  
Received:  
48 65 6C 6C 6F 20 57 6F 72 6C 64 20 23 30 0 20  
20 20 20 0  
Got: Hello World #0  
RSSI: -21  
Sent a reply  
Received:  
48 65 6C 6C 6F 20 57 6F 72 6C 64 20 23 31 0 20  
20 20 20 0  
Got: Hello World #1  
RSSI: -22  
Sent a reply  
Received:  
48 65 6C 6C 6F 20 57 6F 72 6C 64 20 23 32 0 20  
20 20 20 0  
Got: Hello World #2  
RSSI: -21  
Sent a reply
```

☐ Autoscroll Both NL & CR 115200 baud

Subsistema de Comunicaciones

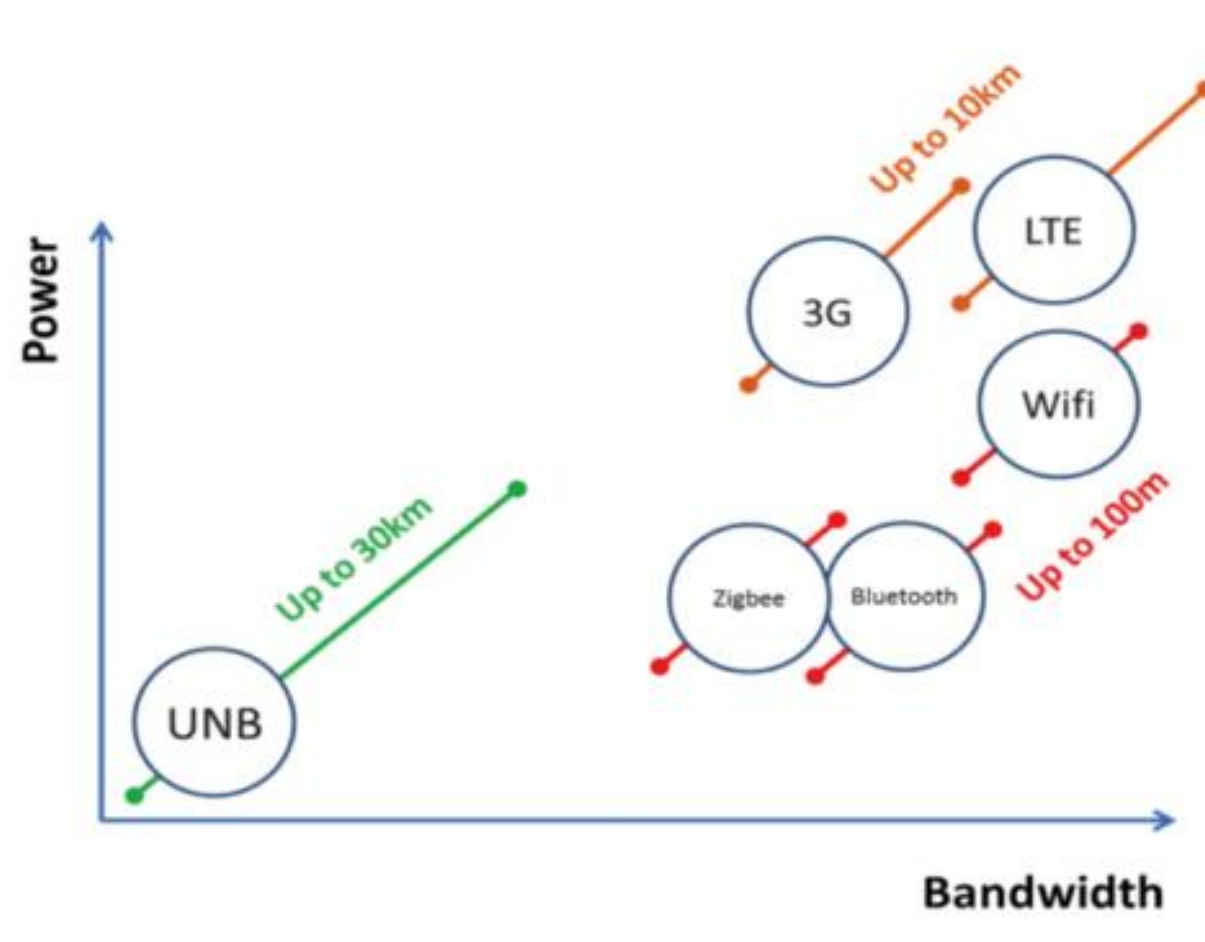
- SigFox.



- SigFox es una compañía francesa fundada en 2009 que proporciona el servicio de red de cobertura amplia de bajo consumo –Low-Power Wide-Area Network (LPWAN)-, es inalámbrica y fue creada para que funcione e interactúe con dispositivos de bajo consumo energético –tales como sensores que funcionan con pilas convencionales- con tasas de transferencias de datos de hasta 12 bytes.
- La red funciona con la tecnología de transmisión UNB –*ultra narrow band*– y consiste en emplear canales estrechos del espectro para alcanzar grandes distancias con un requerimiento mínimo de energía.

Subsistema de Comunicaciones

- SigFox.



Subsistema de Comunicaciones

- SigFox.



- El funcionamiento de esta red es muy similar a las redes de telefonía celular debido a que esta red funciona a partir de la colocación de varias estaciones receptoras y transmisoras; la diferencia entre las estaciones SigFox y las de telefonía celular es en que los dispositivos y sensores que estén conectados a la red SigFox no están sujetas a una sola estación base específica, esto es, cualquier estación puede recibir la información y transmitirla hacia la nube.

Subsistema de Comunicaciones

- SigFox.



- Los servicios principales con los que cuenta esta red son:
 - Sistema de mensajería.
 - Servicio de nube y manejo de datos.
 - Servicios de geolocalización y rastreo.

Subsistema de Comunicaciones

- SigFox




- El [enlace de radio SIGFOX](#) utiliza bandas de radio ISM sin licencia. Las frecuencias exactas pueden variar de acuerdo a las regulaciones nacionales, pero, en **Europa, la banda de 868 MHz** es la más utilizada mientras que en EE UU es la de 915 MHz.
- La densidad de las células en la red SIGFOX se basa en un rango promedio de unos 30-50km en las zonas rurales. En las zonas urbanas, donde hay más obstáculos y el ruido es mayor, la densidad podrá reducirse a entre 3 y 10 km . Las distancias pueden ser mucho mayores para los nodos al aire libre, donde SIGFOX ha conseguido alcances de más de 1.000 kilómetros, lo que la hace especialmente adecuada para zonas despobladas o lejanas.

Subsistema de Comunicaciones

- SigFox



Subsistema de Comunicaciones



Search...

All Categories ▾

Q

HOME

RASPBERRY & ARDUINO PLATFORMS ▾


MODULES - SENSORS ▾

ENCLOSURES ▾

IOT LPWAN NETWORKS ▾

IOT

🏠 > Kit Sigfox Breakout board BRKWS01 + Antenna







Kit Sigfox Breakout board BRKWS01 + Antenna

SigFox Breakout board based on Wisol SFM10R1 module enables easily Sigfox integration in your IOT prototype (1 year subscription included)

Excl. Tax: €19.90 Incl. Tax: **€23.88**

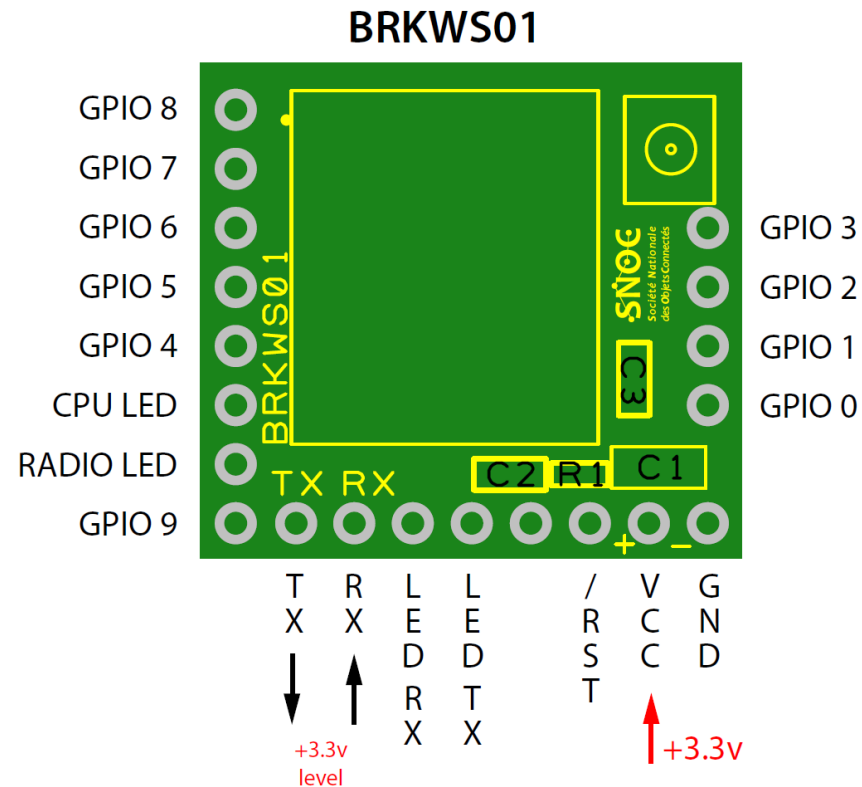
Availability: In stock

[Email to a Friend](#)



Subsistema de Comunicaciones

- SigFox



[COMANDOS AT](#)

Subsistema de Sensorización

- Sensores de distancias.
 - También denominados sensores de proximidad. Miden distancias a objetos, detectan objetos y a partir de un conjunto de sensores, se puede montar un mapa del entorno, mapa de obstáculos con mayor o menor precisión.
 - Existen varios tipos,
 - Interruptores de posición.
 - Capacitivos.
 - Inductivos.
 - Infrarrojos.
 - Ultrasónicos.
 - Magnéticos.

Subsistema de Sensorización

- Interrupciones de posición.
 - También denominados finales de carrera.
 - Son dispositivos eléctricos, neumáticos o mecánicos situados al final del recorrido de un elemento móvil, como por ejemplo una cinta transportadora, con el objetivo de enviar señales que puedan modificar el estado de un circuito. Internamente pueden contener interruptores normalmente abiertos (NA), cerrados (NC).
 - Su uso es muy diverso y se usan en todas las máquinas que tengan un movimiento rectilíneo de ida y vuelta o sigan una trayectoria fija.
 - Aquellas que realizan una carrera o recorrido fijo, como por ejemplo ascensores, montacargas, robots, etc.

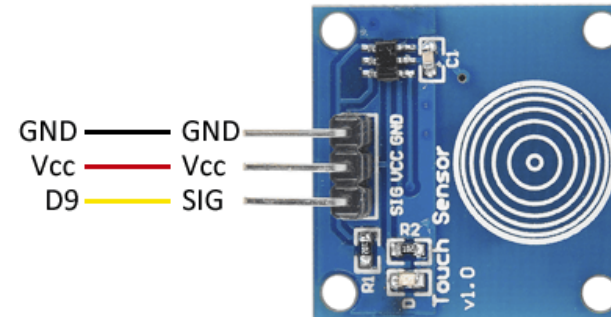
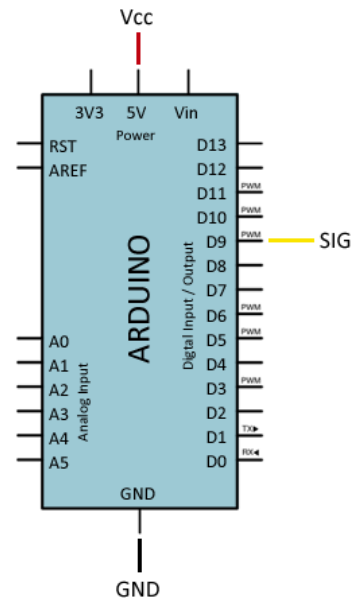
Subsistema de Sensorización



Subsistema de Sensorización

- Sensores capacitivos.
 - Se basa en señalar un cambio de estado, basado en la variación del estímulo de un campo eléctrico.
 - Los sensores capacitivos detectan objetos metálicos, o no metálicos, midiendo el cambio en la capacitancia, la cual depende de la constante dieléctrica del material a detectar, su masa, tamaño, y distancia hasta la superficie sensible del detector.
 - Los detectores capacitivos están contruidos sobre la base de un oscilador LC.
 - Debido a la influencia del objeto a detectar, y del cambio de capacitancia, la amplificación se incrementa haciendo entrar en oscilación el oscilador.
 - Cuando un objeto conductor se acerca a la cara activa del detector, el objeto actúa como un condensador. El cambio de la capacitancia es significativo durante una larga distancia. Si se aproxima un objeto no conductor solamente se produce un cambio pequeño en la constante dieléctrica, y el incremento en su capacitancia es muy pequeño comparado con los materiales conductores.
 - Este detector se utiliza comúnmente para detectar material no metálico: papel, plástico, madera, etc. ya que funciona como un condensador.

Subsistema de Sensorización



```
1  const int sensorPin = 9;
2
3  void setup()
4  {
5      Serial.begin(9600);
6      pinMode(sensorPin, INPUT);
7  }
8
9  void loop()
10 {
11     int estado = digitalRead(sensorPin);
12
13     //mandar mensaje a puerto serie en función del valor leído
14     if (estado == HIGH)
15     {
16         Serial.println("Contacto detectado");
17         //aquí se ejecutarían las acciones
18     }
19     delay(1000);
20 }
```

Subsistema de Sensorización

- Sensores inductivos.
 - El funcionamiento es similar al capacitivo pero en este caso se utiliza una bobina.
 - La bobina detecta el objeto cuando se produce un cambio en el campo electromagnético y envía la señal al oscilador, luego se activa el disparador y finalmente al circuito de salida hace la transición entre abierto o cerrado.

Subsistema de Sensorización

- Sensores infrarrojos.
 - El sensor de infrarrojos es un sensor de medición de distancia, que se basa en un sistema de emisión/recepción de radiación lumínica en el espectro de los infrarrojos (menor que las ondas de radio y mayor que la luz).
 - Una de las técnicas más habituales para la medición de la distancia es mediante la triangulación del haz de luz colimada, si bien también se puede "estimar" la distancia de un objeto a partir de la cantidad de energía recibida tras rebotar la luz sobre un objeto.
 - En robótica móvil se suelen utilizar sensores baratos de corto alcance, en un rango máximo de unos 50/80 cm. y el tipo de detección que realizan es direccional, es decir, sólo son capaces de detectar objetos que están enfrente del sensor.
 - Este tipo de sensor presenta el inconveniente de ser sensible a la luz ambiente como consecuencia de que los rayos de sol también emiten en el espectro de luz infrarroja. Por este motivo, son sensores que se utilizan habitualmente en entornos con iluminación artificial de forma predominante (interiores).

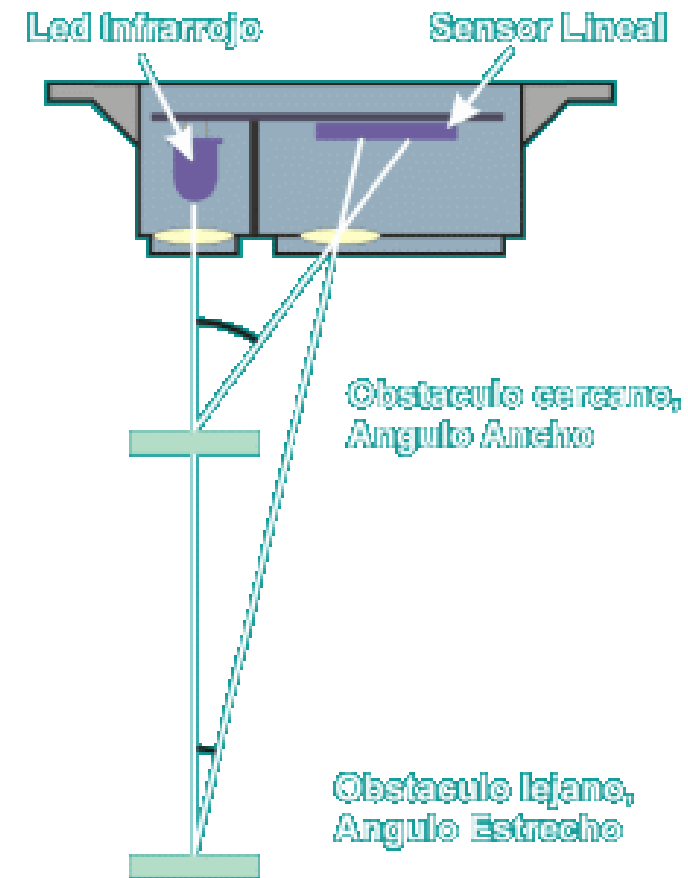
Subsistema de Sensorización

- Sensores infrarrojos.
 - Los sensores más utilizados en robótica son de la marca Sharp, en concreto el modelo GP2D12.



Subsistema de Sensorización

- Sensores infrarrojos.
 - Principio de medición. Triangulación.
 - El sensor infrarrojo funciona mediante el principio de triangulación de la luz que rebota sobre el objeto.
 - El haz de luz incide con un ángulo diferente en función de la distancia del sensor. Este ángulo de incidencia es captado por una película lineal fotosensible que proporciona un valor analógico a la salida en función de la posición en la que el rayo de luz impacta.
 - Se puede fácilmente apreciar que uno de los principales inconvenientes de esta técnica de medición es que el ángulo de incidencia apenas varía para grandes distancias, con lo que el sensor es poco sensible para grandes distancias.



Subsistema de Sensorización

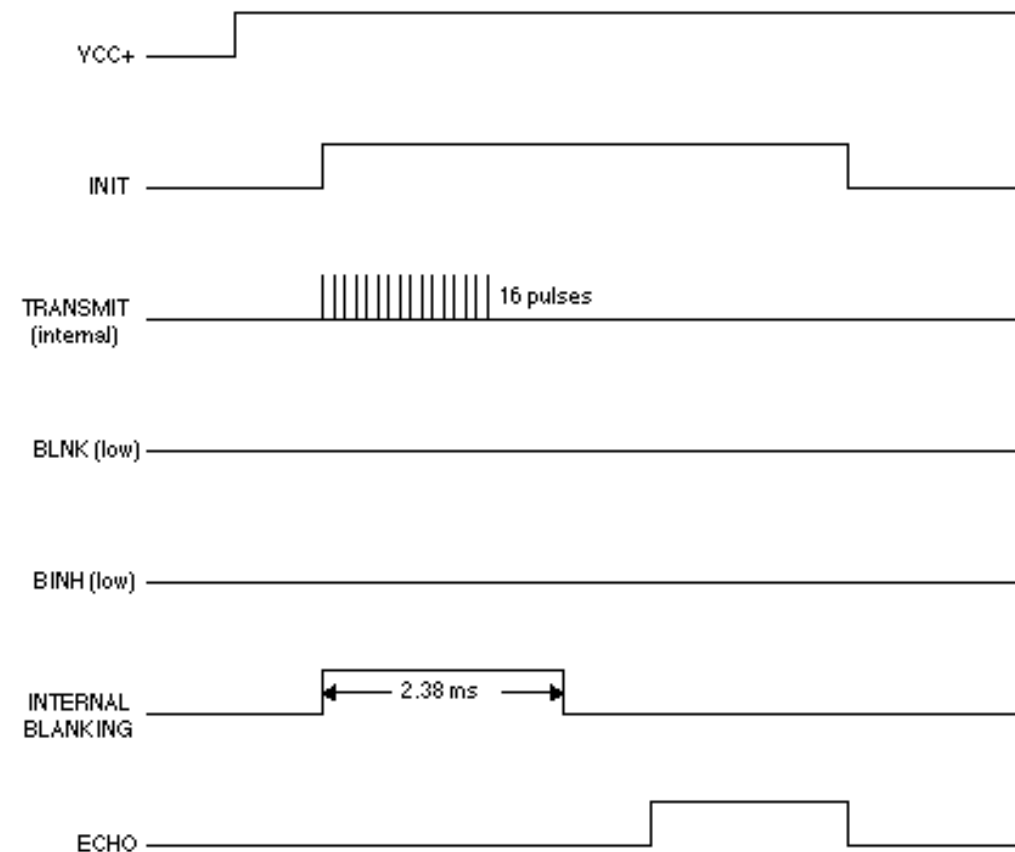
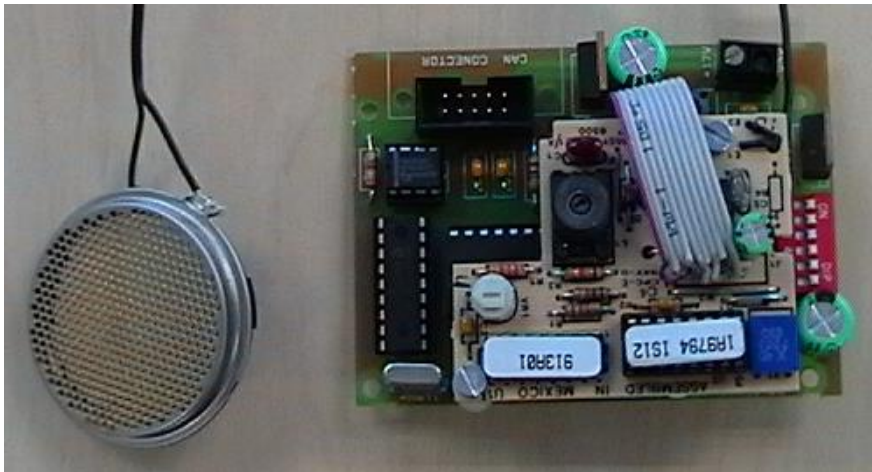
- Sensores ultrasónicos.
 - Los sensores de ultrasonido son muy frecuentes en los robots móviles (particularmente) y de forma significativa en los AUVs, por sus buenas propiedades de medición en entornos acuáticos y sirven para detectar objetos y medir distancias.
 - Se utilizan para construir mapas del entorno y evitar obstáculos.
 - El sensor de ultrasonidos es un dispositivo de medición de distancia que se basa en las propiedades magnetoestrictivas de determinados materiales.
 - Una lámina de material magnetoestrictivo o membrana tiene la propiedad de deformarse mecánicamente y generar ultrasonidos al ser excitada por una corriente eléctrica. El efecto contrario también se produce, es decir, que una vibración mecánica produce una corriente eléctrica. Por tanto, estos sensores emiten una radiación ultrasónica que rebota en los obstáculos del entorno y captan los ecos recibidos.

Subsistema de Sensorización

- Sensores ultrasónicos.
 - Existen dos tipos de configuraciones.
 - **Con sistema de emisor/receptor en el mismo transductor:** Las ondas ultrasónicas se generan con una sola membrana que debe inmediatamente ser "bloqueada" tras emitir el tren de impulsos para poder "escuchar" el rebote de las ondas. Dado que requiere un tiempo de bloqueo de la membrana, esta configuración es, por lo general, no sensible a distancias muy cortas (típicamente pocos centímetros).

Subsistema de Sensorización

- Sensores ultrasónicos.

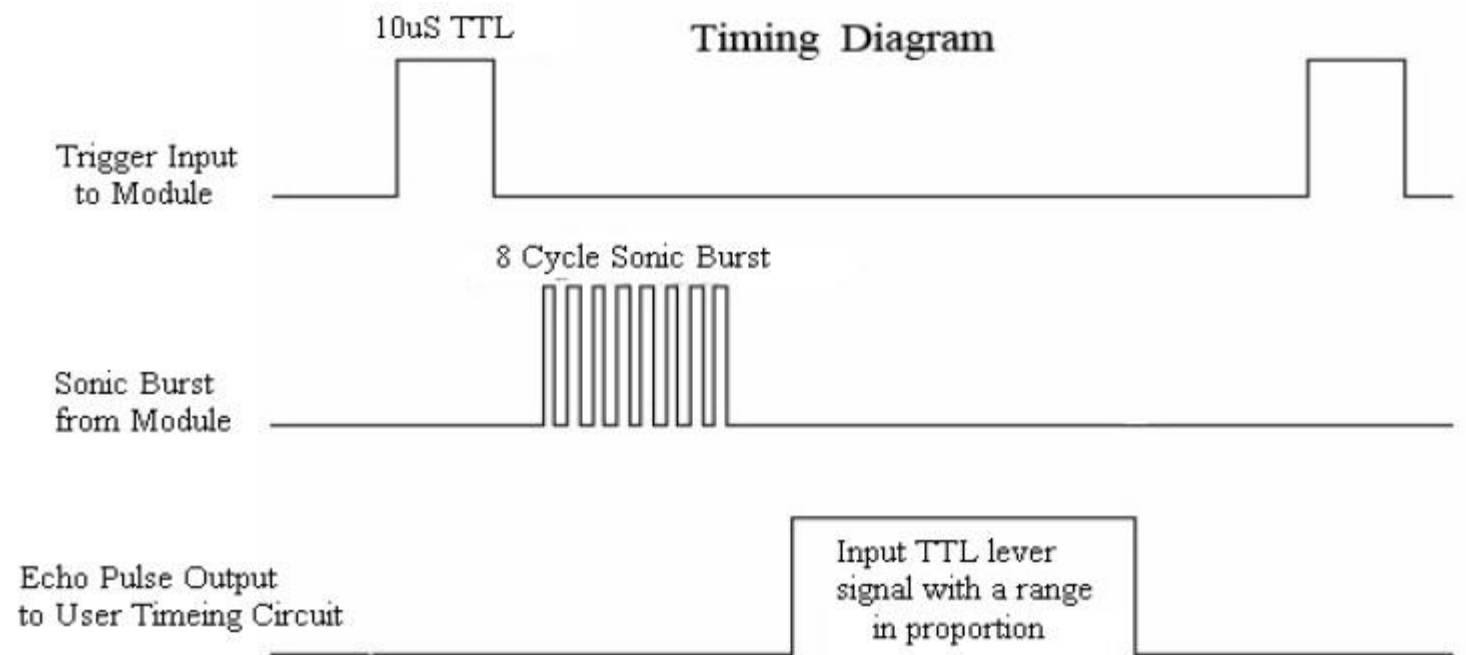


Subsistema de Sensorización

- Sensores ultrasónicos.
 - Existen dos tipos de configuraciones.
 - **Con sistema de emisor/receptor separados:** Las ondas ultrasónicas se generan con una de las membranas, mientras que la otra membrana se utiliza para "escuchar" las ondas ultrasónicas. En esta configuración, la membrana receptora está preparada para escuchar desde el mismo instante en el que se acaban de emitir las ondas y por tanto suele ser apropiada para medir distancias muy cortas que la configuración de un emisor/receptor común no puede alcanzar.

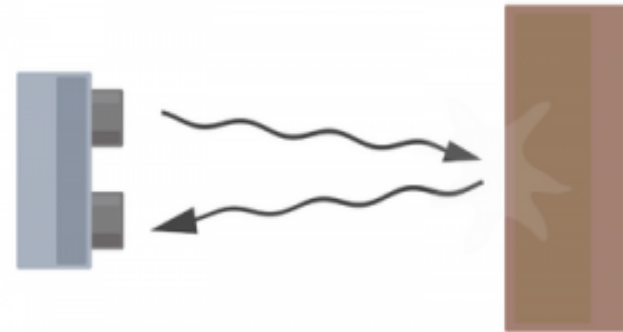
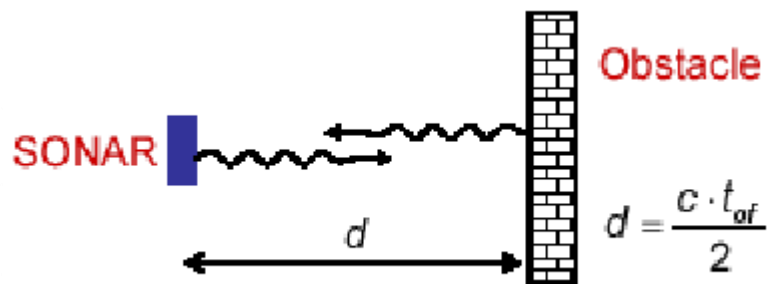
Subsistema de Sensorización

- Sensores ultrasónicos.



Subsistema de Sensorización

- Sensores ultrasónicos.
 - Técnicas de medición.
 - **Tiempo de vuelo:** La técnica consiste en emitir un tren de impulsos y poner un temporizador en marcha. Cuando se recibe el eco de los impulsos emitidos, el tiempo transcurrido es proporcional al doble de la distancia al obstáculo (tiempo de impacto + tiempo de eco). Si en un determinado margen de tiempo no se ha recibido eco, se considera que no hay obstáculo.

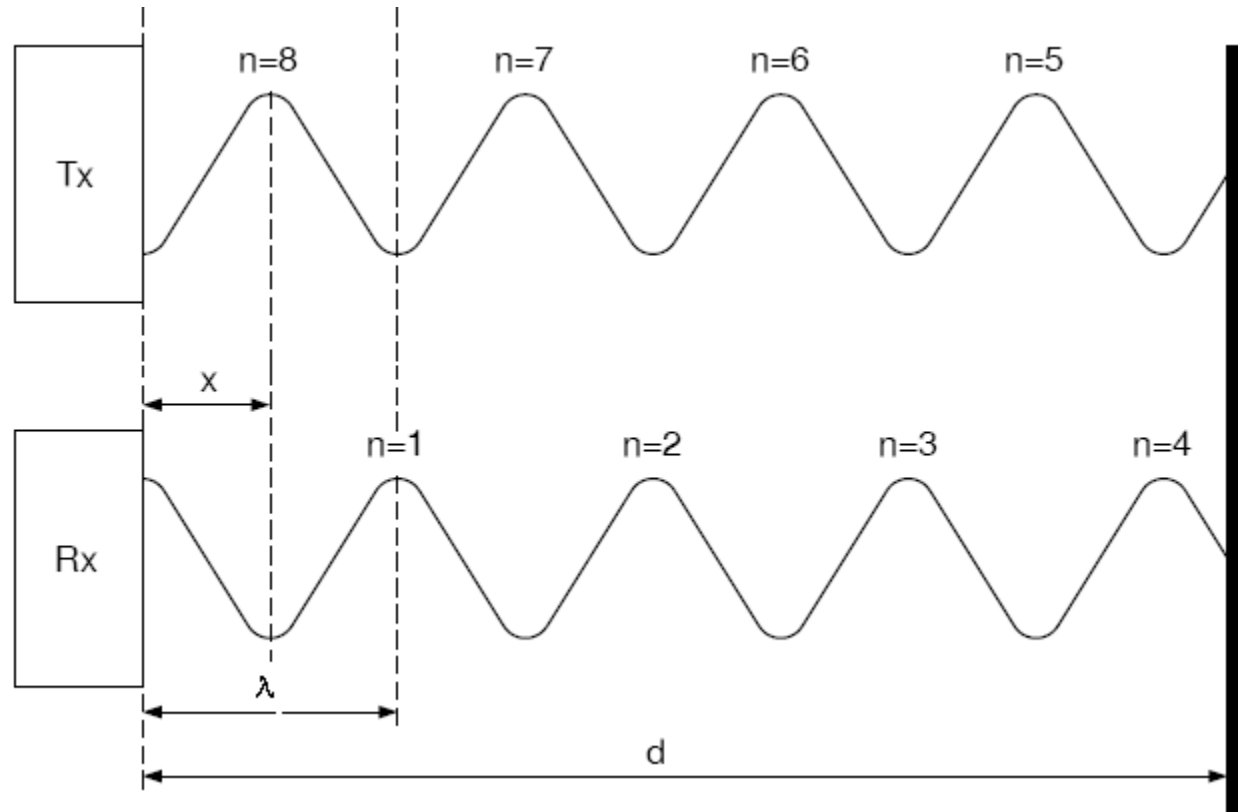


Subsistema de Sensorización

- Sensores ultrasónicos.
 - Técnicas de medición.
 - **Cambio de fase:** Se envía una onda periódica y se reciben los ecos de forma continua. Según el desfase entre las ondas se calcula la distancia. Una característica interesante a remarcar es que el entorno se puede modelar como la convolución entre la señal emitida y la señal de recepción.

Subsistema de Sensorización

- Sensores ultrasónicos.

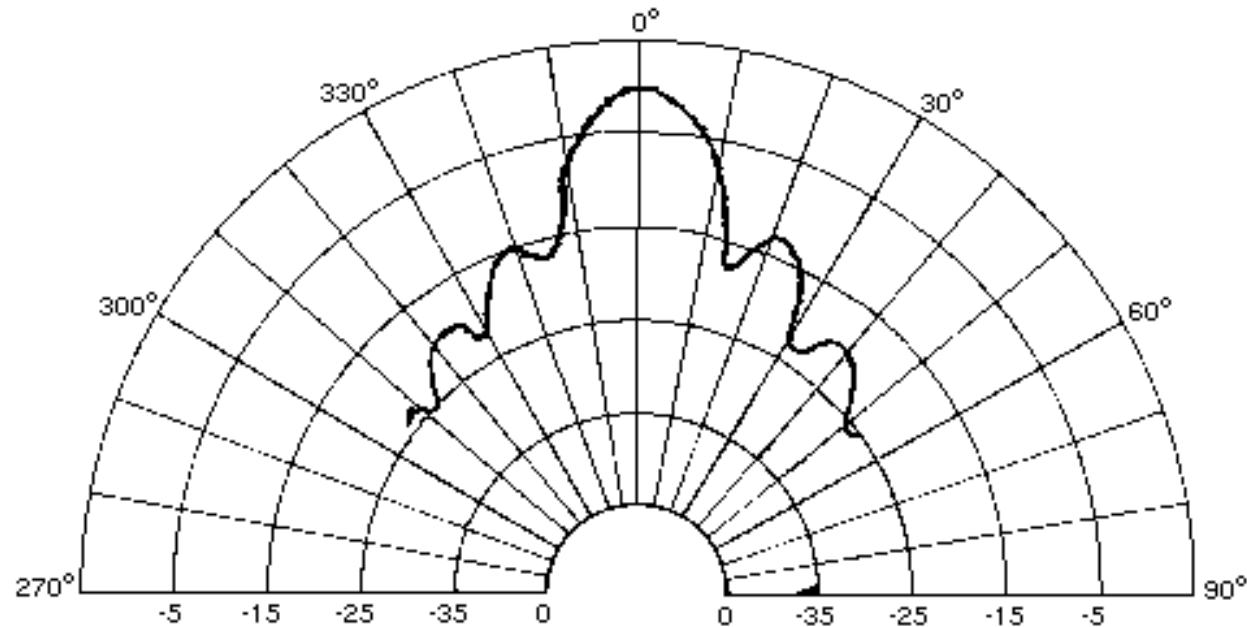


Subsistema de Sensorización

- Sensores ultrasónicos.
 - Propiedades de los sensores de ultrasonidos.
 - Frecuentemente, se utilizan cuatro frecuencias de onda: 60Khz, 56Khz, 52.5Khz y 49Khz.
 - Velocidad del sonido a unos 20°C es de 343.2m/s.
 - La onda ultrasónica tiene un ángulo de detección en el cual es teóricamente sensible y por tanto toda onda que venga en un ángulo dentro del lóbulo principal.
 - También existen otros lóbulos secundarios en los cuales el sensor es también sensible, aunque en menor medida. La hoja de características del sensor nos indica la sensibilidad del mismo y la atenuación de la onda (en dBs) según el ángulo de incidencia.

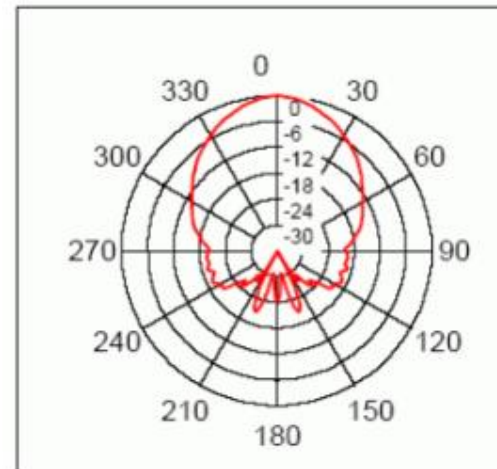
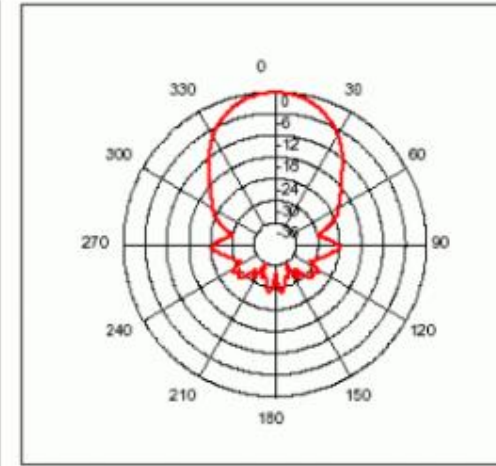
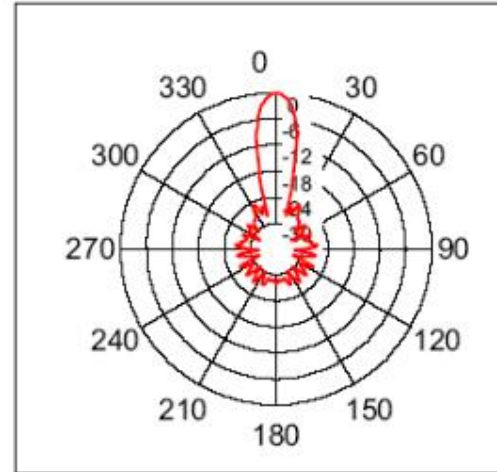
Subsistema de Sensorización

- Sensores ultrasónicos.
 - Propiedades de los sensores de ultrasonidos.



Subsistema de Sensorización

- Sensores ultrasónicos.
 - Propiedades de los sensores de ultrasonidos.

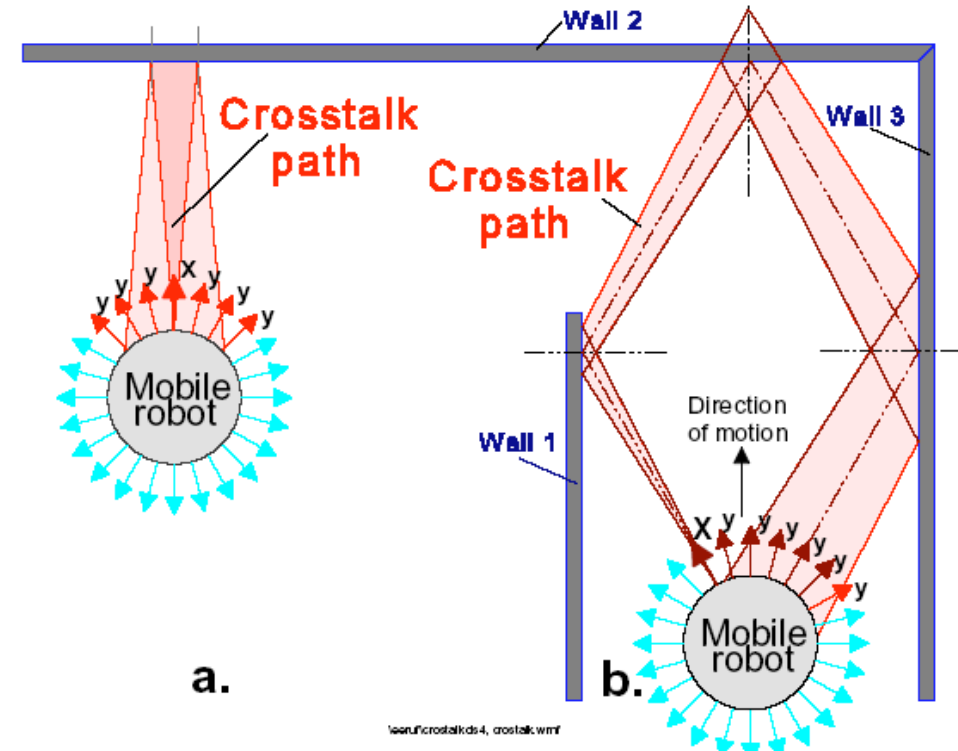
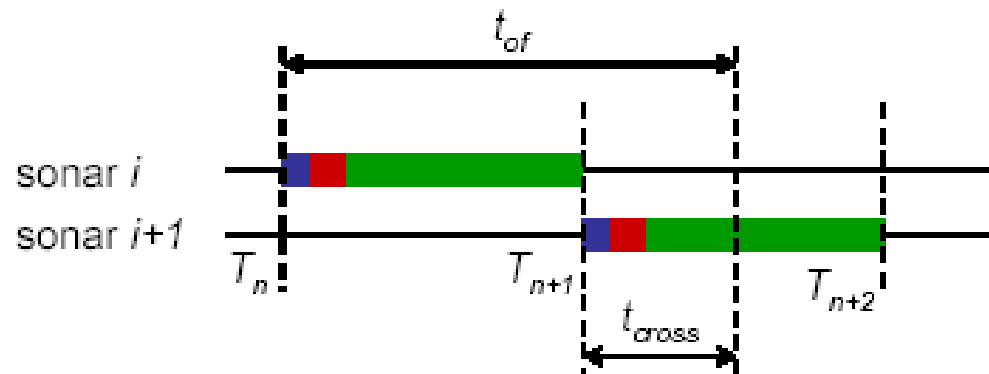


Subsistema de Sensorización

- Sensores ultrasónicos.
 - Propiedades de los sensores de ultrasonidos.
 - Los sensores tienen una distancia máxima de detección, que depende en gran medida de la frecuencia de la onda ultrasónica, la "sensibilidad" de la electrónica y la membrana y por supuesto el medio de transmisión.
 - Si se trata del aire, la onda se degrada rápidamente en comparación de medios acuáticos.
 - Para el caso de los ultrasonidos en el aire podemos encontrar sensores con distancias máximas de 3 o 4 metros, por lo general menores de 10m.
 - Los sensores pueden verse afectados por el fenómeno cross-talking: la onda de un sensor es recibida por otro sensor o un mismo sensor puede recibir su propia onda de un disparo previo si los tiempos de espera entre disparo y disparo no son adecuados.
 - Si tenemos varios sensores de ultrasonidos y queremos minimizar el efecto cross-talking, lo lógico es secuenciar los disparos, aún así será muy difícil distinguir entre los ecos recibidos por un robot u otro en el caso de que utilicen los mismos sensores.

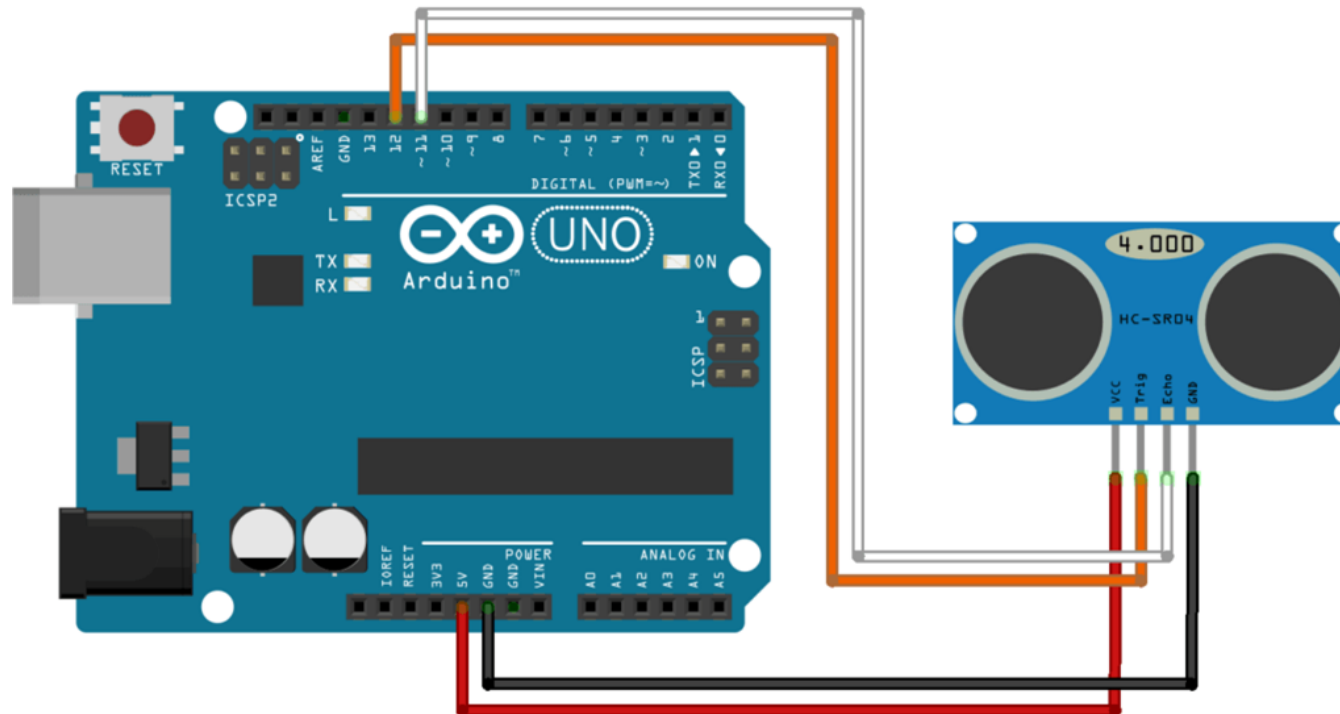
Subsistema de Sensorización

- Sensores ultrasónicos.
 - Propiedades de los sensores de ultrasonidos.



Subsistema de Sensorización

- Sensores ultrasónicos.
 - <http://playground.arduino.cc/Code/NewPing>



Subsistema de Sensorización

- Sensores ultrasónicos.

```
1  #include <NewPing.h>
2
3  /*Aqui se configuran los pines donde debemos conectar el sensor*/
4  #define TRIGGER_PIN 12
5  #define ECHO_PIN    11
6  #define MAX_DISTANCE 200
7
8  /*Crear el objeto de la clase NewPing*/
9  NewPing sonar(TRIGGER_PIN, ECHO_PIN, MAX_DISTANCE);
10
11 void setup() {
12     Serial.begin(9600);
13 }
14
15 void loop() {
16     // Esperar 1 segundo entre mediciones
17     delay(1000);
18     // Obtener medicion de tiempo de viaje del sonido y guardar en variable uS
19     int uS = sonar.ping_median();
20     // Imprimir la distancia medida a la consola serial
21     Serial.print("Distancia: ");
22     // Calcular la distancia con base en una constante
23     Serial.print(uS / US_ROUNDTRIP_CM);
24     Serial.println("cm");
25 }
```

Subsistema de Sensorización

- Sensores ultrasónicos.
 - Sin librería.

```
1  /**
2   GeekFactory - "Construye tu propia tecnologia"
3   Distribucion de materiales para el desarrollo e innovacion tecnologica
4   www.geekfactory.mx
5
6   EJEMPLO SENSOR ULTRASONICO 1
7
8   REALIZA LA MEDICION DE DISTANCIA CON UN SENSOR ULTRASÓNICO HC-SR04 CONECTADO
9   AL ARDUINO. ESTE EJEMPLO NO UTILIZA LIBRERIAS ESPECIALES PARA EL SENSOR, SOLAMENTE
10  SE UTILIZA LA FUNCION "PULSEIN" PARA MEDIR LA LONGITUD DEL PULSO DE ECO.
11
12  */
13  // DECLARACION DE VARIABLES PARA PINES
14  const int pinecho = 8;
15  const int pintrigger = 9;
16  const int pinled = 13;
17
18  // VARIABLES PARA CALCULOS
19  unsigned int tiempo, distancia;
20
21  void setup() {
22    // PREPARAR LA COMUNICACION SERIAL
23    Serial.begin(9600);
24    // CONFIGURAR PINES DE ENTRADA Y SALIDA
25    pinMode(pinecho, INPUT);
26    pinMode(pintrigger, OUTPUT);
27    pinMode(13, OUTPUT);
28  }
29
30  void loop() {
31    // ENVIAR PULSO DE DISPARO EN EL PIN "TRIGGER"
32    digitalWrite(pintrigger, LOW);
33    delayMicroseconds(2);
34    digitalWrite(pintrigger, HIGH);
35    // EL PULSO DURA AL MENOS 10 uS EN ESTADO ALTO
```

Subsistema de Sensorización

- Sensores ultrasónicos.
 - Sin librería.

```
36  delayMicroseconds(10);
37  digitalWrite(pintrigger, LOW);
38
39  // MEDIR EL TIEMPO EN ESTADO ALTO DEL PIN "ECHO" EL PULSO ES PROPORCIONAL A LA DISTANCIA MEDIDA
40  tiempo = pulseIn(pinecho, HIGH);
41
42  // LA VELOCIDAD DEL SONIDO ES DE 340 M/S O 29 MICROSEGUNDOS POR CENTIMETRO
43  // DIVIDIMOS EL TIEMPO DEL PULSO ENTRE 58, TIEMPO QUE TARDA RECORRER IDA Y VUELTA UN CENTIMETRO LA ONDA
44  distancia = tiempo / 58;
45
46  // ENVIAR EL RESULTADO AL MONITOR SERIAL
47  Serial.print(distancia);
48  Serial.println(" cm");
49  delay(200);
50
51  // ENCENDER EL LED CUANDO SE CUMPLA CON CIERTA DISTANCIA
52  if (distancia <= 15) {
53      digitalWrite(13, HIGH);
54      delay(500);
55  } else {
56      digitalWrite(13, LOW);
57  }
58 }
```

Subsistema de Sensorización

- Sensores ultrasónicos.
 - Sin librería.

`pulseIn()`

[Advanced I/O]

Description

Reads a pulse (either **HIGH** or **LOW**) on a pin. For example, if **value** is **HIGH**, `pulseIn()` waits for the pin to go from **LOW** to **HIGH**, starts timing, then waits for the pin to go **LOW** and stops timing. Returns the length of the pulse in microseconds or gives up and returns 0 if no complete pulse was received within the timeout.

The timing of this function has been determined empirically and will probably show errors in longer pulses. Works on pulses from 10 microseconds to 3 minutes in length.

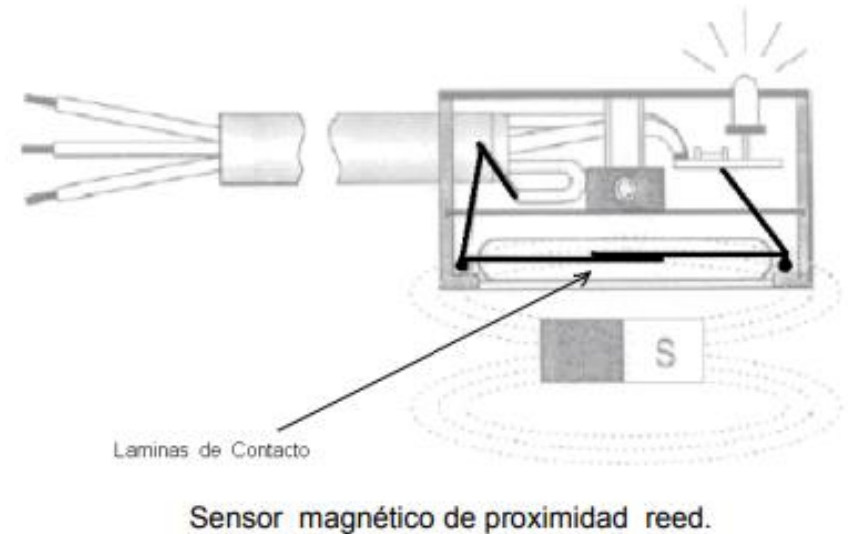
Syntax

```
pulseIn(pin, value)
```

```
pulseIn(pin, value, timeout)
```

Subsistema de Sensorización

- Sensores Magnéticos.
 - Estos sensores de proximidad, reaccionan ante los campos magnéticos de imanes permanentes y de electroimanes.
 - En el caso de un sensor reed, las láminas de contactos están hechas de material ferromagnéticos y están sellados dentro de un pequeño tubo de vidrio. El tubo se llena con un gas inerte, comúnmente nitrógeno.



Subsistema de Sensorización

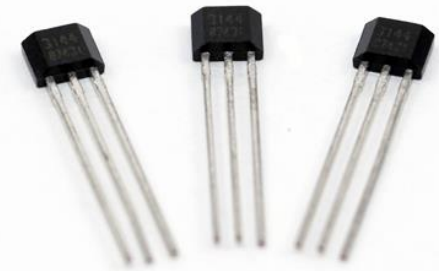
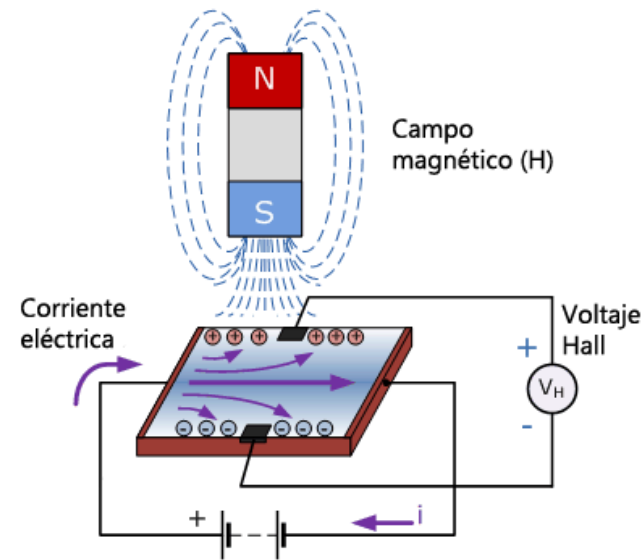
- Sensores Magnéticos.
 - Si se acerca un campo magnético al sensor de proximidad, las láminas se unen por magnetismo y se produce un contacto eléctrico.
 - Por otro lado, nos encontramos con los sensores de efecto hall. Un sensor Hall es un dispositivo que nos permite realizar mediciones de campo magnético.
 - Una ventaja importante de los sensores Hall es que **realizan la medición a distancia**, sin necesidad de contacto físico. Aunque su alcance es limitado (típicamente pocos centímetros) esto supone que apenas presentan desgaste mecánico. Además **son inmunes a ruidos y polvo**. Esto los convierte en sensores fiables y duraderos.

Subsistema de Sensorización

- Sensores Magnéticos.
 - En general, encontramos dos tipos de sensores Hall,
 - **Analógicos.** Generan una salida proporcional a la intensidad del campo magnético. Empleados para medir la intensidad de un campo magnético.
 - **Digitales.** Proporcionan un valor Alto en presencia de campo magnético, y Bajo en ausencia del mismo. Por tanto, son empleados para detectar la existencia de campos magnéticos. A su vez se dividen en,
 - Switch, se activan al acercar el polo, y se desactivan al retirar el polo.
 - Latch, se activan al acercar un polo, y mantienen su valor hasta que se acerca un polo contrario.
 - **Ejemplo: Sensor Hall A3144, de tipo digital Switch.**
 - Podemos emplear este sensor para detectar la presencia de un objeto, al que previamente habremos colocado un pequeño imán, o para fabricar tacómetros (contadores de revoluciones) simplemente acoplado un pequeño imán de neodimio al eje.

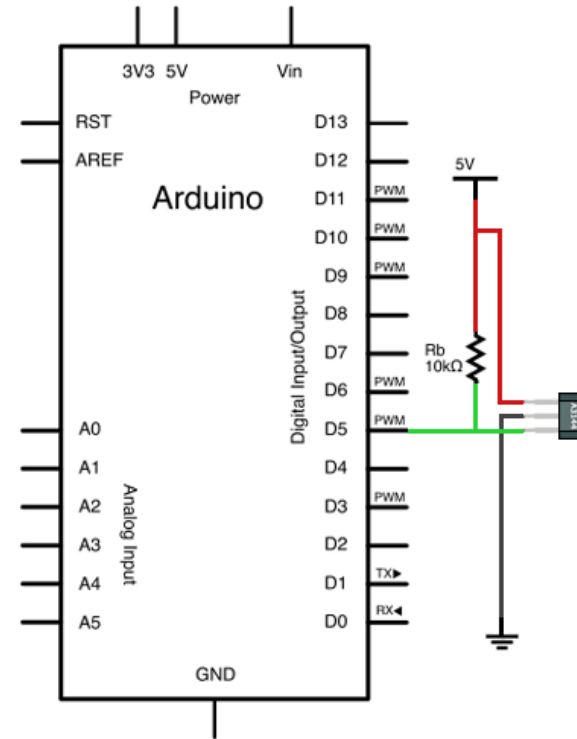
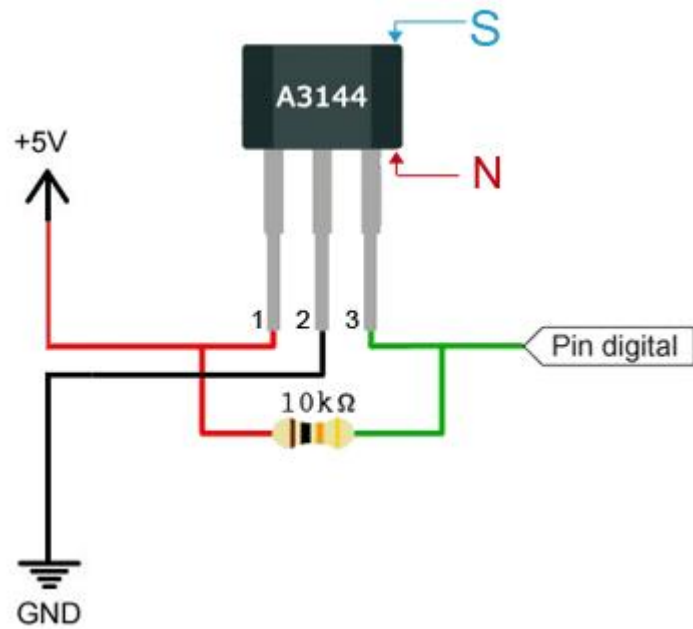
Subsistema de Sensorización

- Sensores Magnéticos.
 - Midiendo esta tensión originada por el efecto Hall podemos conseguir construir sensores y medidores de campos magnéticos.



Subsistema de Sensorización

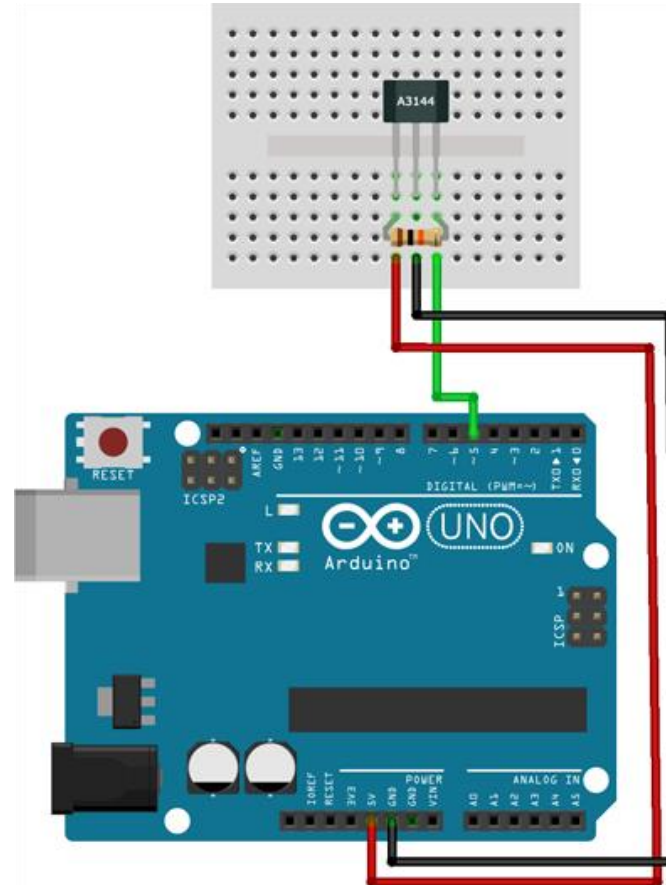
- Sensores Magnéticos.



Subsistema de Sensorización

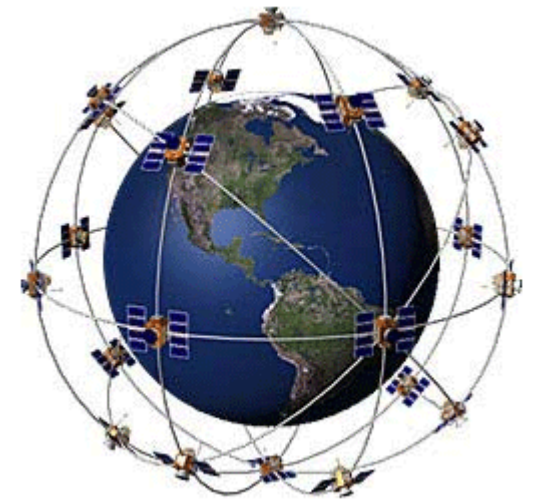
- Sensores Magnéticos.

```
1  const int HALLPin = 5;
2  const int LEDPin = 13;
3
4  void setup() {
5      pinMode(LEDPin, OUTPUT);
6      pinMode(HALLPin, INPUT);
7  }
8
9  void loop() {
10     if(digitalRead(HALLPin)==HIGH)
11     {
12         digitalWrite(LEDPin, HIGH);
13     }
14     else
15     {
16         digitalWrite(LEDPin, LOW);
17     }
18 }
```



Subsistema de Sensorización

- Sensores de posicionamiento absoluto – GPS
 - El **Sistema de Posicionamiento Global** (en inglés, **GPS**; **Global Positioning System**) es un sistema que permite determinar en toda la Tierra la posición de un objeto con una precisión de hasta centímetros.
 - El GPS funciona mediante una red de como mínimo 24 satélites en órbita sobre el planeta Tierra, a 20 180 km de altura, con trayectorias sincronizadas para cubrir toda la superficie de la Tierra.
 - Cuando se desea determinar la posición tridimensional, el receptor que se utiliza para ello localiza automáticamente como mínimo cuatro satélites de la red, de los que recibe unas señales indicando la identificación y hora del reloj de cada uno de ellos, además de información sobre la constelación.



Subsistema de Sensorización

- Sensores de posicionamiento absoluto – GPS
 - Con base en estas señales, el aparato sincroniza el reloj del GPS y calcula el tiempo que tardan en llegar las señales al equipo, midiendo la distancia al satélite mediante el método de trilateración inversa (triangulación), el cual se basa en determinar la distancia de cada satélite al punto de medición.
 - Conocidas las distancias, se determina fácilmente la propia posición relativa respecto a los satélites.
 - Conociendo además las coordenadas o posición de cada uno de ellos por la señal que emiten, se obtiene la posición absoluta o coordenadas reales del punto de medición.
 - También se consigue una gran exactitud en el reloj del GPS, similar a la de los relojes atómicos que lleva a bordo cada uno de los satélites.

Subsistema de Sensorización

- Sensores de posicionamiento absoluto – GPS

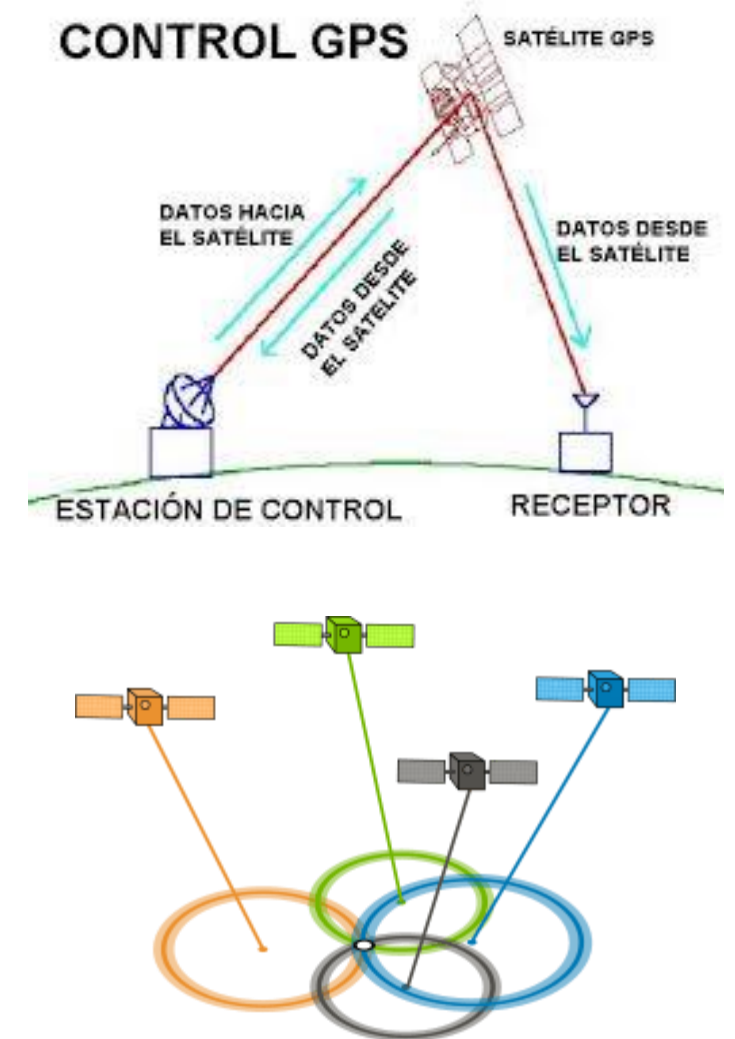


Subsistema de Sensorización

- Sensores de posicionamiento absoluto – GPS
 - La antigua Unión Soviética construyó un sistema similar llamado GLONASS, ahora gestionado por la Federación Rusa.
 - La Unión Europea desarrolló el sistema de navegación Galileo. En diciembre de 2016 la Comisión Europea, propietaria del sistema, informó que el sistema de navegación Galileo comenzó sus operaciones y que los satélites ya envían información de posicionamiento, navegación y determinación de la hora a usuarios de todo el mundo.
 - La República Popular China está implementando su propio sistema de navegación, el denominado Beidou, que está previsto que cuente con 12 y 14 satélites entre 2011 y 2015. Para 2020, ya plenamente operativo deberá contar con 30 satélites. En diciembre de 2012 tenían 14 satélites en órbita.

Subsistema de Sensorización

- Cómo es la señal GPS?
 - Cada satélite GPS emite continuamente un mensaje de navegación a 50 bits por segundo en la frecuencia transportadora de microondas de aproximadamente 1.600 MHz, más concretamente, todos los satélites emiten a 1575,42 MHz (esta es la señal L1) y 1227,6 MHz (la señal L2).
 - La información que es útil al receptor GPS para determinar su posición se llama efemérides. En este caso cada satélite emite sus propias efemérides, en la que se incluye la salud del satélite, su posición en el espacio, su hora atómica, información doppler, etc.
 - **¿Por qué varían los relojes de los satélites?**
 - Retrasan unos 7 μ s/día.



Subsistema de Sensorización

- Sensores de posicionamiento absoluto – GPS
 - El DGPS (Differential GPS), o GPS diferencial, es un sistema que proporciona a los receptores de GPS correcciones de los datos recibidos de los satélites GPS, con el fin de proporcionar una mayor precisión en la posición calculada.
 - El fundamento radica en el hecho de que los errores producidos por el sistema GPS afectan por igual (o de forma muy similar) a los receptores situados próximos entre sí. Los errores están fuertemente correlacionados en los receptores próximos.
 - Un receptor GPS fijo en tierra (referencia) que conoce exactamente su posición basándose en otras técnicas, recibe la posición dada por el sistema GPS, y puede calcular los errores producidos por el sistema GPS, comparándola con la suya, conocida de antemano.
 - Este receptor transmite la corrección de errores a los receptores próximos a él, y así estos pueden, a su vez, corregir también los errores producidos por el sistema dentro del área de cobertura de transmisión de señales del equipo GPS de referencia.
 - Precisión de 1 metro.

Subsistema de Sensorización

- GPS – El protocolo NMEA.
 - La National Marine Electronics Association (NMEA) es una organización de comercio electrónico estadounidense que establece estándares de comunicación entre electrónica marina.
 - El protocolo NMEA es adoptado para la comunicación entre el receptor GPS y un microcontrolador.
 - La sentencia NMEA de un loran empieza por LC y la de un GPS por GP.
 - Algunos FORMATOS DE SENTENCIAS SOPORTADOS,
 - GGA - Global Positioning System Fix Data
 - GLC - Geographic Position, Loran-C
 - GLL - Geographic Position, Latitude/Longitude
 - GSA - GPS DOP and Active Satellites

Subsistema de Sensorización

1.1 GGA - Global Positioning System Fix Data

Time, position and fix related data for a GPS receiver.

Example:

```
$GPGGA,114353.000,6016.3245,N,02458.3270,E,1,10,0.81,35.2,M,19.5,M,,*50
```

Format:

```
$GPGGA,hhmmss.dd,xxmm.dddd,<N|S>,yyymm.dddd,<E|W>,v,ss,d.d,h.h,M,g.g,M,a.a,xxxx*hh<CR><LF>
```

Subsistema de Sensorización

Type	Description
hhmmss.dd	UTC time of the fix. hh=hours; mm=minutes; ss=seconds; dd=decimal part of seconds
xxmm.dddd	Latitude coordinate. xx=degrees; mm=minutes; dddd=decimal part of minutes
<N/S>	Character denoting either N=North or S=South.
yyymm.dddd	Longitude coordinate. yyy=degrees; mm=minutes; dddd=decimal part of minutes
<E/W>	Character denoting either E=East or W=West.
v	Fix valid indicator 1 = GPS fix (SPS) 2 = DGPS fix 3 = PPS fix 4 = Real Time Kinematic 5 = Float RTK 6 = estimated (dead reckoning) (2.3 feature) 7 = Manual input mode 8 = Simulation mode
ss	Number of satellites used in position fix, 00-12. Notice: Fixed length field of two letters.
d.d	HDOP - Horizontal Dilution Of Precision.
h.h	Altitude (mean-sea-level, geoid)
M	Letter M.
g.g	Difference between the WGS-84 reference ellipsoid surface and the mean-sea-level altitude.
M	Letter M.
a.a	-
xxxx	-

Subsistema de Sensorización

1.5 RMC - Recommended Minimum Specific GNSS Data.

Time, date, position, course and speed data.



Message start as "\$GNRMC" in IT530M default mode both GPS and Glonass constellation.

Example:

```
$GPRMC,114353.000,A,6016.3245,N,02458.3270,E,0.01,0.00,121009,,,A*69
```

Example IT530M hybrid mode:

```
$GNRMC,105440.000,A,6012.5669,N,02449.6536,E,0.00,0.00,061112,,,D*70
```

Format:

```
$GPRMC,hhmmss.dd,S,xxmm.dddd,<N|S>,yyymm.dddd,<E|W>,s.s,h.h,ddmmyy,d.d, <E|W>,M*hh<CR><LF>
```

Subsistema de Sensorización

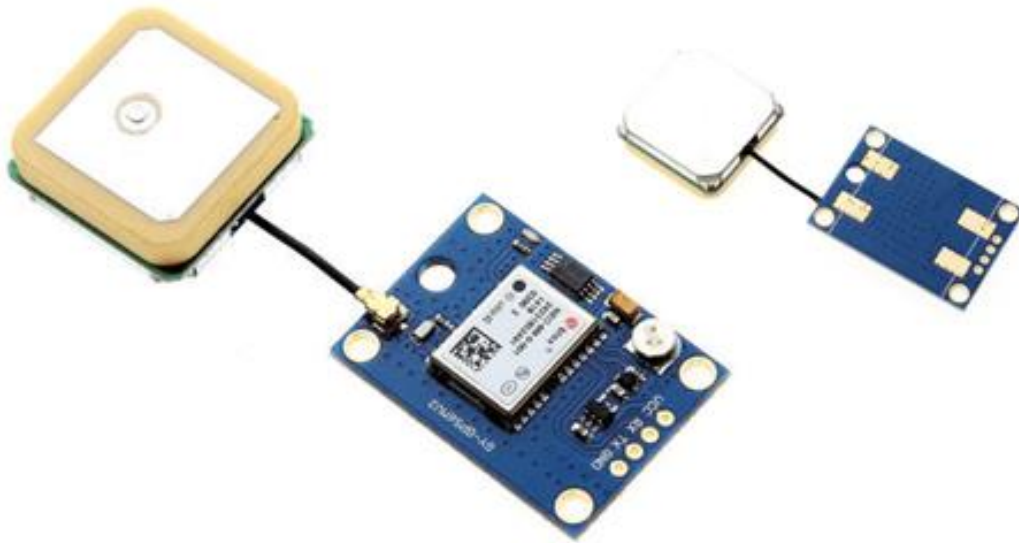
Format:

\$GPRMC,hhmmss.dd,S,xxmm.dddd,<N|S>,yyymm.dddd,<E|W>,s.s,h.h,ddmmyy,d.d, <E|W>,M*hh<CR><LF>

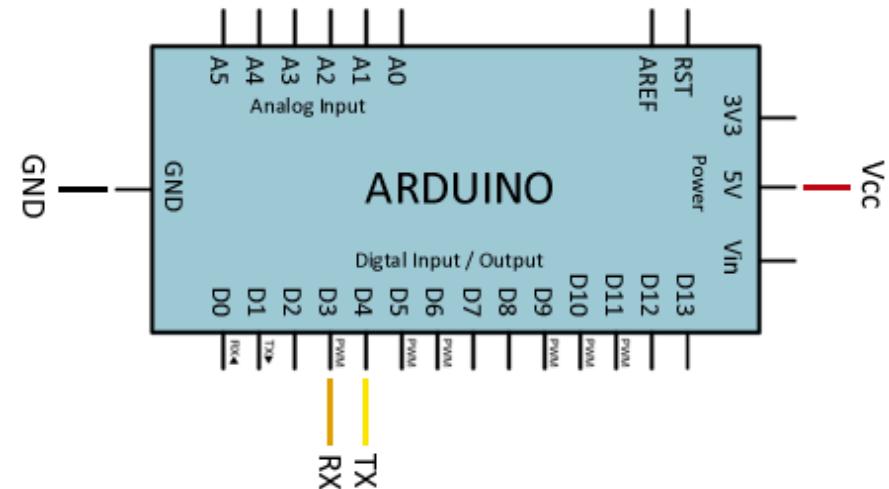
Type	Description
hhmmss.dd	UTC time of the fix. hh=hours; mm=minutes; ss=seconds; dd=decimal part of seconds
S	Status indicator. A=valid; V=invalid
xxmm.dddd	Latitude coordinate. xx=degrees; mm=minutes; dddd=decimal part of minutes
<N S>	Character denoting either N=North or S=South.
yyymm.dddd	Longitude coordinate. yyy=degrees; mm=minutes; dddd=decimal part of minutes
<E W>	Character denoting either E=East or W=West.
s.s	Speed in knots.
h.h	Heading
ddmmyy	UTC Date of the fix. dd=day of month; mm=month; yy=year
d.d	Magnetic variation in degrees, not supported
<E W>	Letter denoting direction of magnetic variation. Either E=East or W=West. Not supported
M	Mode indicator A=autonomous; N=data not valid

Subsistema de Sensorización

- GPS – Arduino
 - Ejemplo: Módulo NEO6MV2



GND — GND
(D4) RX — TX
(D3) TX — RX
5V — Vcc



Subsistema de Sensorización

- GPS – Arduino
 - Ejemplo: Módulo NEO6MV2
 - Los datos obtenidos del NEO-6 están en forma secuencias disponibles en el protocolo NMEA.

```
$GPRMC,hhmmss.ss,A,llll.ll,a,yyyy.yy,a,x.x,x.x,ddmmyy,x.x,a*hh
```

```
1  #include <SoftwareSerial.h>
2
3  const int RX = 4;
4  const int TX = 3;
5
6  SoftwareSerial gps(RX, TX);
7
8  void setup()
9  {
10     Serial.begin(115200);
11     gps.begin(9600);
12 }
13
14 void loop()
15 {
16     if (gps.available())
17     {
18         char data;
19         data = gps.read();
20         Serial.print(data);
21     }
22 }
```

Subsistema de Sensorización

- GPS – Arduino

```
$GPRMC,hhmmss.ss,A,IIII.II,a,yyyy.yy,a,x.x,x.x,ddmmyy,x.x,a*hh
```

hhmmss.ss	Hora UTC
A	Estado receptor (A = OK, V = warning)
IIII.II,a	Latitud (a = N o S)
yyyy.yy,a	Longitud (a = E o W)
x.x	Velocidad en nudos in knots
x.x	Curso en grados
ddmmyy	Fecha UT
x.x,a	Variacion magnética en grados (a = E o W)
*hh	Checksum

Subsistema de Sensorización

- GPS – Arduino

```
$GPRMC,225446,A,4916.45,N,12311.12,W,000.5,054.7,191194,020.3,E*68
```

225446	Hora 22:54:46 UTC
A	Estado receptor A = OK
4916.45,N	Latitud 49° 16.45 min Norte
12311.12,W	Longitud 123°11.12 min Oeste
000.5	Velocidad 0.5 nudos
054.7	Curso 54.7°
191194	Fecha 19 Noviembre 1994
020.3,E	Variación magnética 20.3° East
*68	Checksum

Subsistema de Sensorización

- GPS – Arduino
 - El “parseo” de datos puede hacerse manualmente o bien mediante una librería preparada para ello.
 - Librería Arduino: TinyGPS.

Subsistema de Sensorización

- Sensores de posicionamiento de motores.
 - La posición de los motores de un robot puede controlarse mediante tres técnicas principales aunque existen algunas más dependiendo del tipo de motor.
 - En general, para motores de corriente continua normales nos encontramos con estas técnicas,
 - Sensorización de la posición mediante encoders relativos.
 - Sensorización de la posición mediante encoders absolutos.
 - Técnica del PWM.

Subsistema de Sensorización

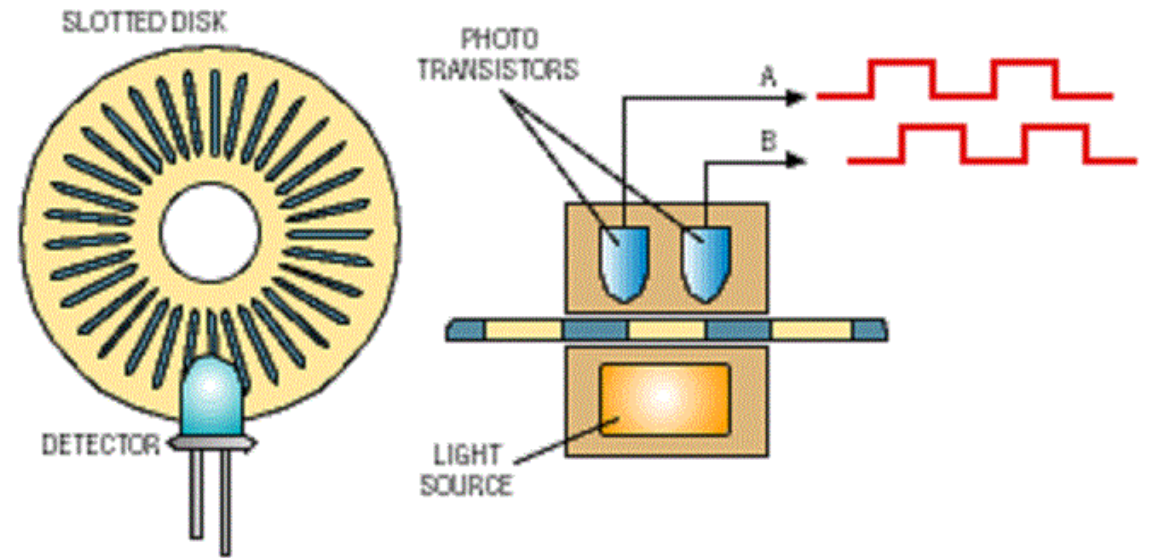
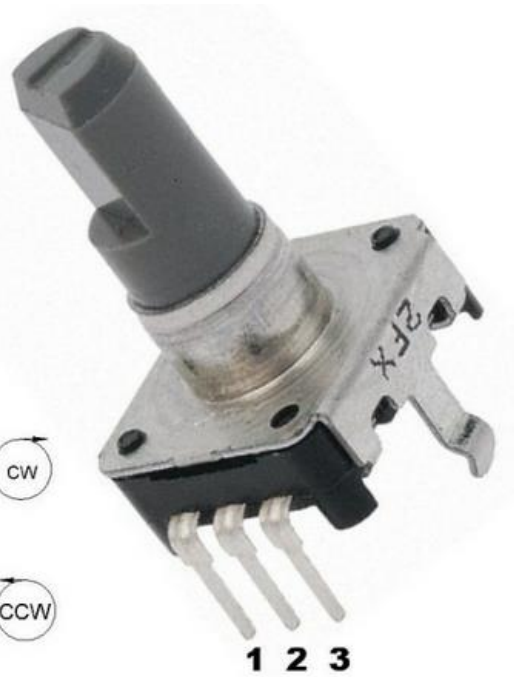
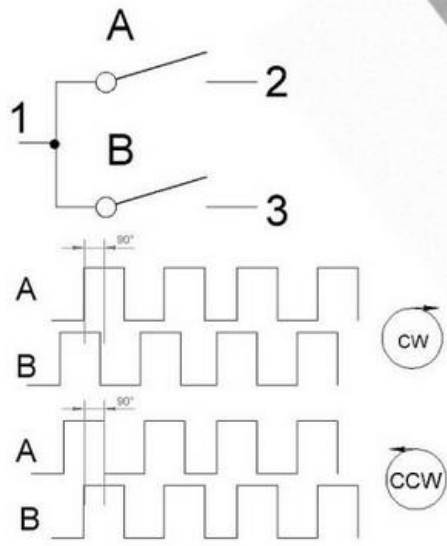
- Sensores de posicionamiento de motores.
 - Encoders relativos.
 - En el caso de un encoder de posición relativa, no tiene una posición inicial o final, sino que puede girar libremente en cualquier sentido.
 - Lo único que hace es entregar una serie de impulsos cuando se desplaza.
 - Estos pulsos varían en función del sentido de giro, por lo que un sistema digital decodifica estos impulsos para convertirlos en una magnitud manejable, como distancia o grados de giro.
 - Los encoders más comunes son las ruedecillas que leen la posición de un ratón de bola.



Subsistema de Sensorización

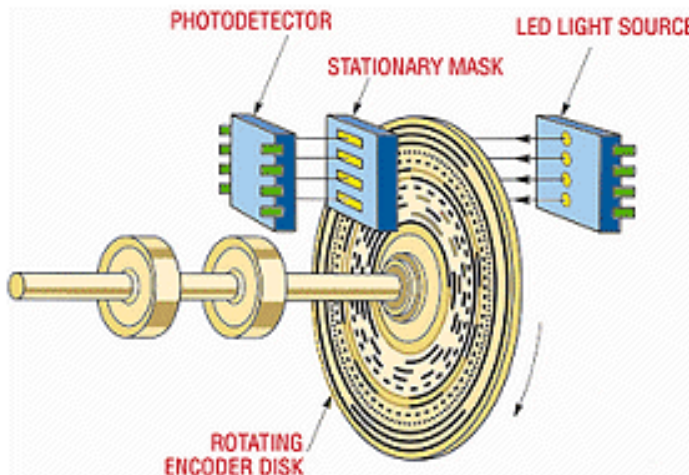
- Sensores de posicionamiento de motores.
 - Encoders relativos.

Encoder Rotativo



Subsistema de Sensorización

- Sensores de posicionamiento de motores.
 - Encoders absolutos.
 - En todo momento sabemos la posición en la que está el motor girando.
 - Se lleva a cabo mediante un sistema numérico codificado.
 - No es necesario saber el estado anterior para determinar el actual.
 - Son más caros que los relativos / incrementales.

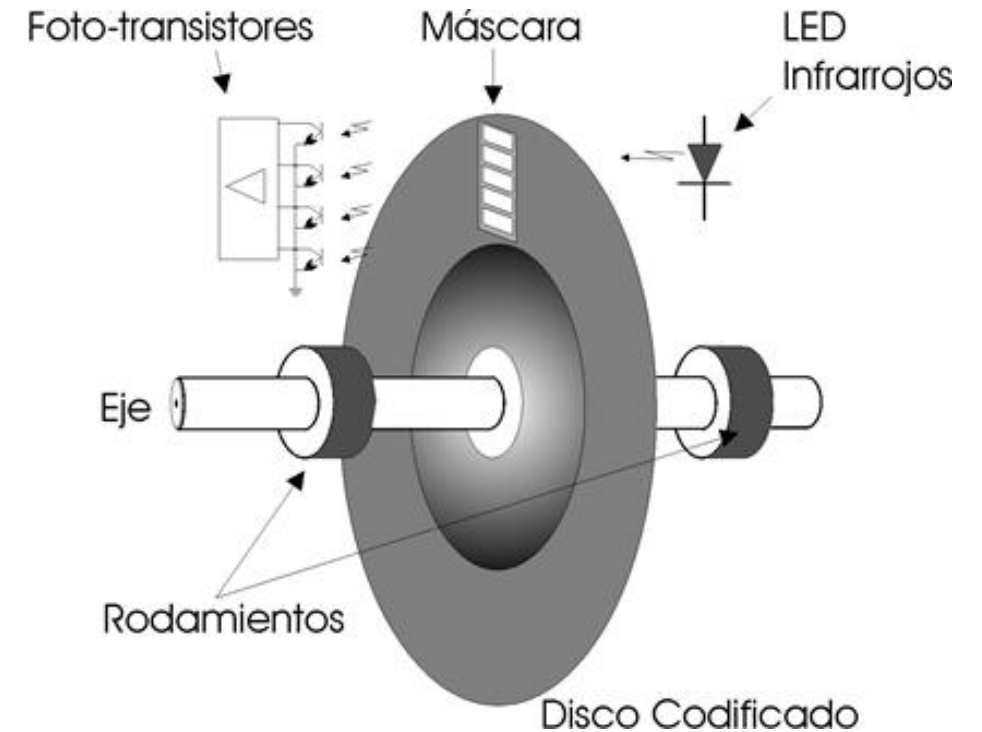
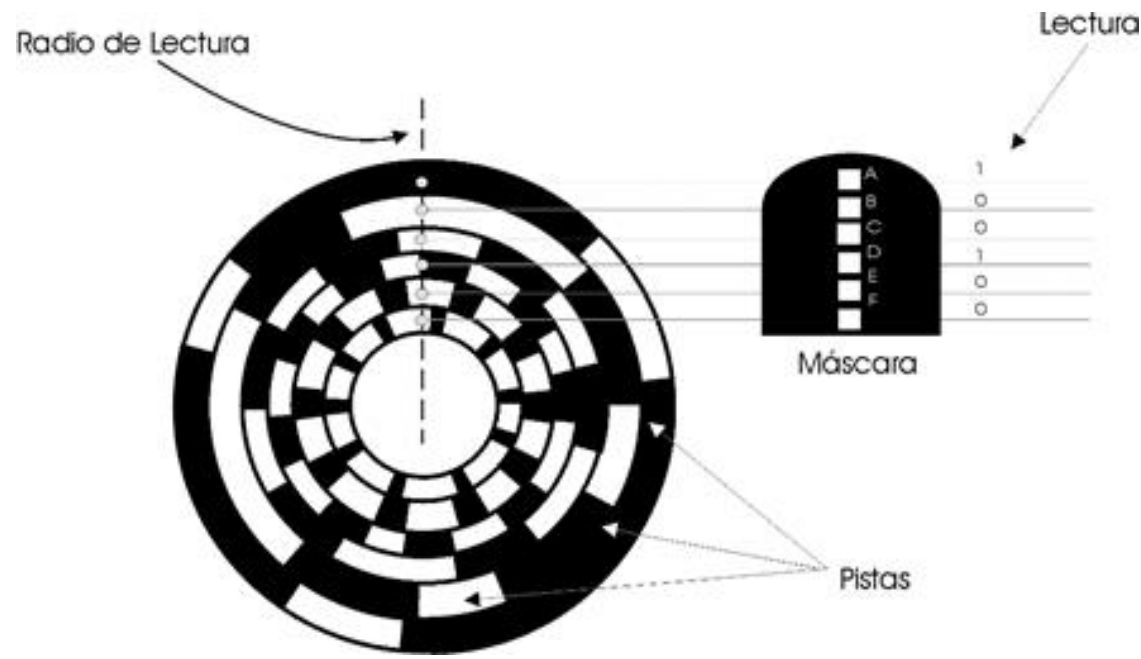


Absolute Encoder



Subsistema de Sensorización

- Sensores de posicionamiento de motores.
 - Encoders absolutos.



Subsistema de Sensorización

- Sensores de posicionamiento de motores.
 - Técnica del PWM.
 - Hablamos de la función *PWM* **como abreviatura de la modulación por ancho de pulsos**, algo que se ha convertido en una práctica habitual de los interruptores de potencia modernos, controlando la energía de inercia.
 - Esta acción tiene en cuenta la modificación del proceso de trabajo de una señal de tipo periódico.
 - Puede tener varios objetivos, como tener el control de la energía que se proporciona a una carga o llevar a cabo la transmisión de datos.
 - El ciclo de trabajo de una señal periódica es el ancho relativo de su parte positiva en relación con el período.
 - Expresado matemáticamente:

$$D = r/T$$

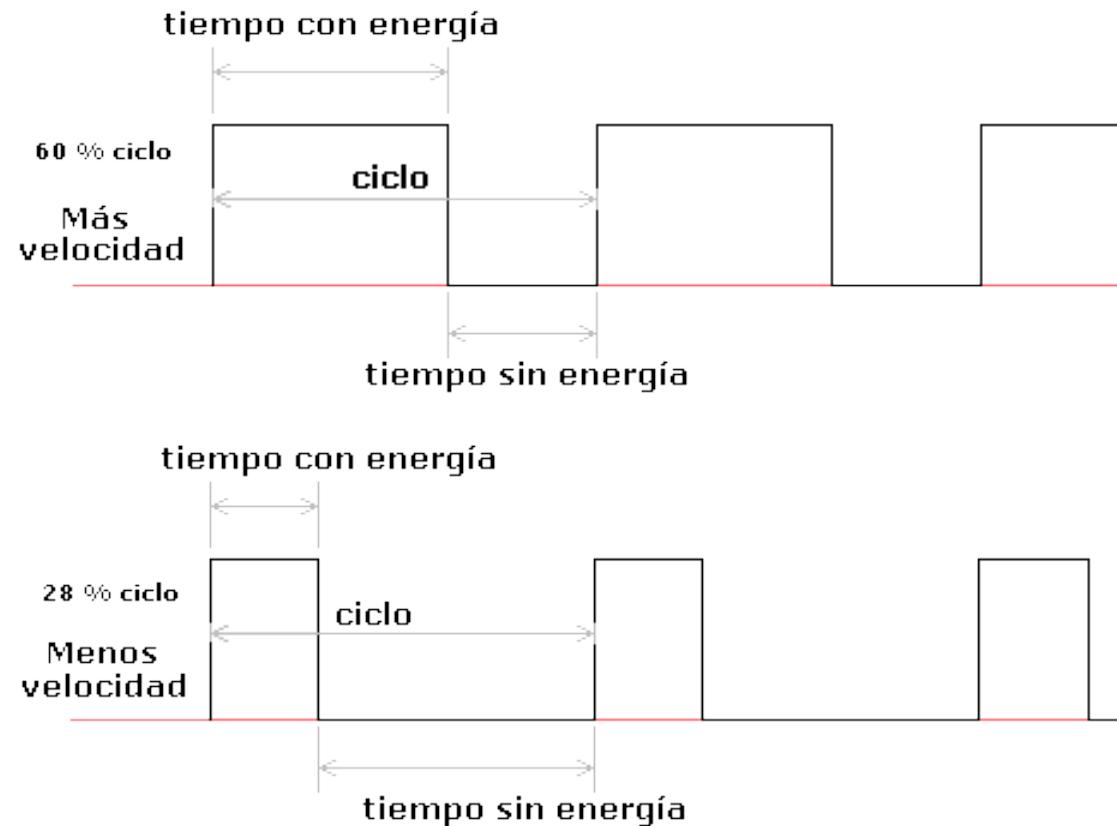
D es el ciclo de trabajo (duty cycle).

r es el tiempo en que la función es positiva (ancho del pulso).

T es el período de la función.

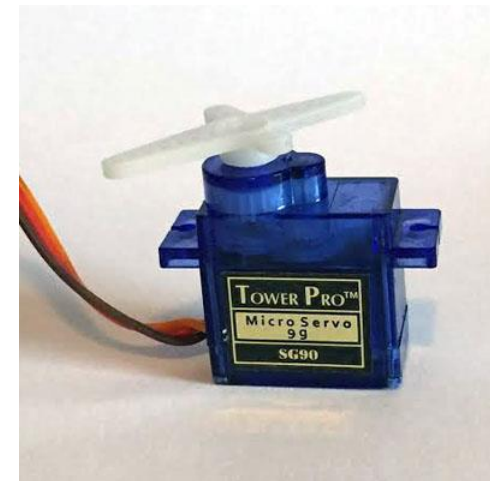
Subsistema de Sensorización

- Sensores de posicionamiento de motores.
 - Técnica del PWM.



Subsistema de Sensorización

- Sensores de posicionamiento de motores.
 - Técnica del PWM.
 - Se puede controlar la posición de un servomotor mediante la técnica del PWM.
 - Un servomotor es un motor eléctrico pero con dos características especiales.
 - Por un lado, nos permite **mantener la posición** que indiquemos, siempre que esté dentro del rango de operación del propio dispositivo.
 - Por otro lado nos permite **controlar la velocidad de giro**, podemos hacer que antes de que se mueva a la siguiente posición espere un tiempo.

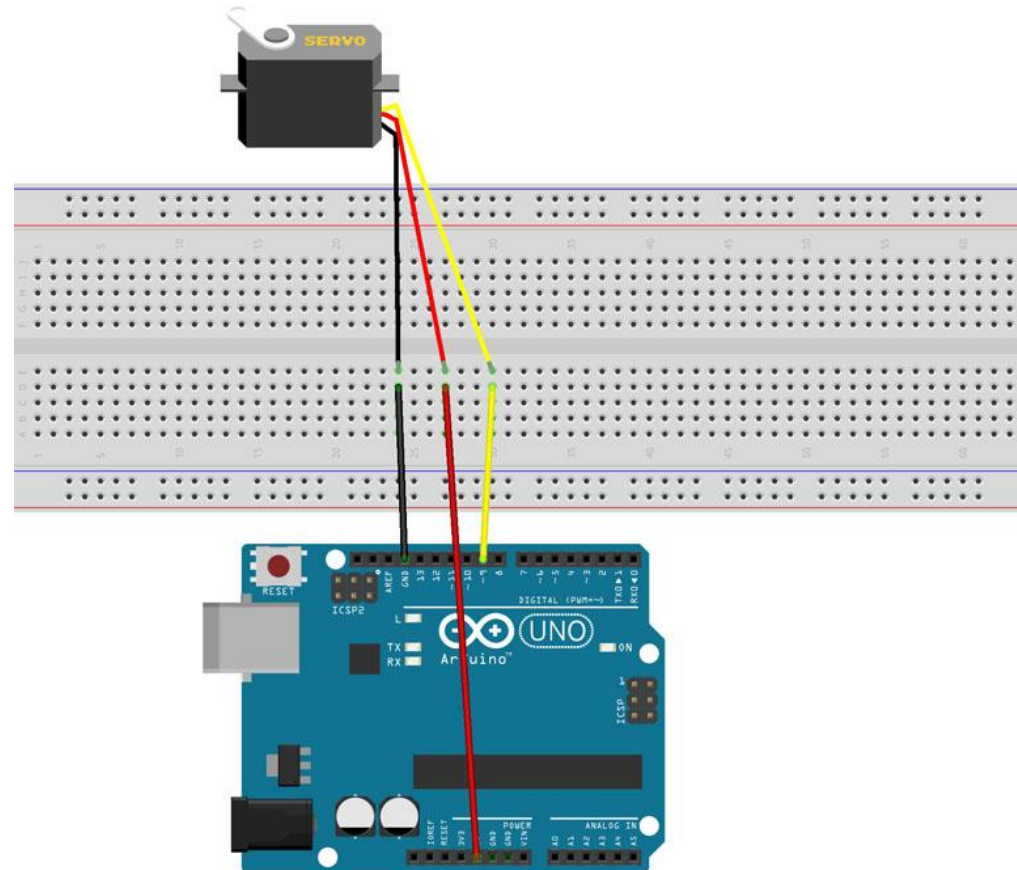


Subsistema de Sensorización

- Sensores de posicionamiento de motores.
 - Técnica del PWM.
 - El ángulo de giro, en este caso nos permite hacer un **barrido entre -90° y 90°**. Lo que viene a ser un **ángulo de giro de 180°**.
 - Aunque el servo puede moverse con una resolución de más de 1 grado, este es el máximo de resolución que vamos a conseguir debido a la limitación de la señal PWM que es capaz de generar el microcontrolador de Arduino.
 - Estos motores funcionan con una señal **PWM**, con un pulso de trabajo entre 1 ms y 2 ms y con un periodo de 20 ms (50 Hz).
 - ¿Qué quiere decir todo esto? Este dato nos indica la **velocidad máxima** a la que podemos mover el servomotor con Arduino. Sólo podremos cambiar de posición cada 20 ms. Esto dependerá del tipo y marca de nuestro servo.

Subsistema de Sensorización

- Sensores de posicionamiento de motores.
 - Técnica del PWM.



Subsistema de Sensorización

- Sensores de posicionamiento de motores.
 - Código para girar el motor de 0º a 180º.
 - Librería *Servo.h*

```
2 // Creado: Luis del Valle (luisvalleprograma@gmail.com)
3 // https://programarfacil.com
4 */
5
6 // Incluimos la librería para poder controlar el servo
7 #include <Servo.h>
8
9 // Declaramos la variable para controlar el servo
10 Servo servoMotor;
11
12 void setup() {
13     // Iniciamos el monitor serie para mostrar el resultado
14     Serial.begin(9600);
15
16     // Iniciamos el servo para que empiece a trabajar con el pin 9
17     servoMotor.attach(9);
18 }
19
20 void loop() {
21
22     // Desplazamos a la posición 0º
23     servoMotor.write(0);
24     // Esperamos 1 segundo
25     delay(1000);
26
27     // Desplazamos a la posición 90º
28     servoMotor.write(90);
29     // Esperamos 1 segundo
30     delay(1000);
31
32     // Desplazamos a la posición 180º
33     servoMotor.write(180);
34     // Esperamos 1 segundo
35     delay(1000);
36 }
```

Subsistema de Sensorización

- Sensores de posicionamiento de motores.

- Girando grado a grado el servomotor

```
9 // Declaramos la variable para controlar el servo
10 Servo servoMotor;
11
12 void setup() {
13     // Iniciamos el monitor serie para mostrar el resultado
14     Serial.begin(9600);
15
16     // Iniciamos el servo para que empiece a trabajar con el pin 9
17     servoMotor.attach(9);
18
19     // Inicializamos al ángulo 0 el servomotor
20     servoMotor.write(0);
21 }
22
23 void loop() {
24
25     // Vamos a tener dos bucles uno para mover en sentido positivo y otro en sentido negativo
26     // Para el sentido positivo
27     for (int i = 0; i <= 180; i++)
28     {
29         // Desplazamos al ángulo correspondiente
30         servoMotor.write(i);
31         // Hacemos una pausa de 25ms
32         delay(25);
33     }
34
35     // Para el sentido negativo
36     for (int i = 179; i > 0; i--)
37     {
38         // Desplazamos al ángulo correspondiente
39         servoMotor.write(i);
40         // Hacemos una pausa de 25ms
41         delay(25);
42     }
43 }
```