# Capítulo 3 – Arduino

En este capítulo conoceremos qué es la plataforma Arduino, sus comienzos y otras tecnologías que colaboraron en el desarrollo de la misma. Además, analizaremos características de la placa, examinando capacidades técnicas como el microcontrolador, memoria y medios de comunicación.

También veremos el abanico de placas producidas por la compañía, sus especificaciones técnicas, similitudes y diferencias. Por otro lado, se examinarán diversos sensores, actuadores y módulos compatibles con la plataforma Arduino. Por último, se comentará la aplicación en las instituciones educativas y la utilización de Arduino en el SAR.

## **3.1 Arduino**

Arduino es una plataforma y compañía, del mismo nombre, de electrónica "open-source" o de código abierto cuyo objetivo es brindar hardware y software de fácil utilización. Es decir, se propone como una plataforma sencilla con una curva de aprendizaje baja para realizar proyectos interactivos para público no necesariamente con conocimientos técnicos.

Ilustración - Logo Arduino

Arduino se trata de una SBC (AGREGAR REFERENCIA CAPITULO 2.4) con entradas y salidas, analógicas y digitales, la cual es programada bajo un entorno de desarrollo, basado en el entorno de programación inspirado en **Processing** y en la estructura de programación **Wiring**. En la imagen (Ilustración 1 - Logo Arduino - Logo Arduino) se puede ver el logo oficial de la compañía.

## **3.2 Historia**

Arduino se inició en el año 2005 como un proyecto para estudiantes en el Instituto IVREA, en Ivrea (Italia). Dado que se utilizaba el microcontrolador BASIC Stamp, cuyo costo era alto para para los fines educativos se comienza el proyecto Arduino. El nombre del proyecto viene del nombre del Bar di Re Arduino (Bar del Rey Arduino), donde Massimo Banzi empezaba a desarrollarlo. [1]

Un estudiante, Hernando Barragán, quien desarrolló la tarjeta electrónica Wiring, el lenguaje de programación y la plataforma de desarrollo. ​Una vez concluida dicha plataforma, los investigadores trabajaron para hacerlo más ligero, más económico y de mayor alcance a la comunidad de hardware y código abierto.

Posteriormente, Google colaboró en el desarrollo del Kit Android ADK (Accesory Development Kit), una placa Arduino capaz de comunicarse directamente con teléfonos móviles inteligentes bajo el sistema operativo Android para que el teléfono controle luces, motores y sensores conectados a Arduino. ​

Para la producción en serie de la primera versión se tomó en cuenta que el coste no fuera mayor de 30 euros, que fuera ensamblado en una placa de color azul, debía ser Plug and Play y que trabajara con todas las plataformas informáticas tales como MacOSX, Windows y GNU/Linux.

### 3.2.1 Wiring

Wiring es una plataforma de prototipado electrónico de fuente abierta compuesta de un lenguaje de programación, un entorno de desarrollo integrado (IDE), y un microcontrolador.

Esta plataforma permite escribir software para controlar dispositivos conectados a la tarjeta electrónica para crear toda clase de objetos interactivos, espacios o experiencias físicas que sienten y responden al mundo físico.

Este proceso se llama *sketching* con hardware; se explora una gran cantidad de ideas de forma muy rápida, se seleccionan las más interesantes, se afinan y producen prototipos en un proceso iterativo.

Wiring toma de Processing la IDE y el concepto de *sketch*, pero enfocado en la programación de microcontroladores en vez de programación gráfica. Provee una librería de C/C++ la cual simplifica operaciones comunes como el manejo de entrada/salida. Los programas de Wiring están escritos en C/C++, pese a que sus usuarios sólo necesiten definir dos funciones para hacer un programa ejecutable:

setup() – una función ejecutada sólo una vez en el arranque de la placa, la cual puede ser usada para definir los ajustes iniciales de un entorno.

loop() – una función llamada repetidamente hasta que la placa es apagada.

Como podemos apreciar en la siguiente ilustración (Ilustración 2 – Código de Blink en Wiring IDE) hacer un blink a un led es muy sencillo dado la abstracción que nos otorga la librería. Un blink es un parpadeo de un led conectado a la placa. Se lo considera el “hola mundo” de Arduino.

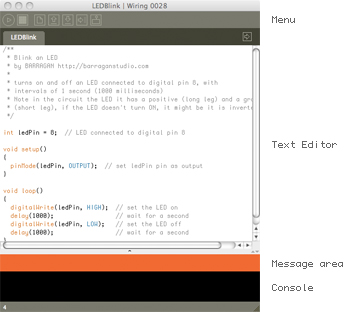


Ilustración – Código de Blink en Wiring IDE

Para ejemplificar la interfase de programación que poreve Wiring al usuario en contrasposición a la utilización de la API del fabricante pude obserbarse un como ejmplo el código (Ilustración 3 - C++ Blink ejemplo) puede ser escrito como Ilustración 2 - Wiring IDE[[1]](#endnote-1)



Ilustración - C++ Blink ejemplo

### 3.2.2 Processing

Es un lenguaje de programación y entorno de desarrollo integrado de código abierto basado en Java, de fácil utilización, y que sirve como medio para la enseñanza y producción de proyectos multimedia e interactivos de diseño digital. En la imagen (Ilustración 4 - Logo de Processing) se puede apreciar su logo.

Uno de los objetivos expresos de Processing es el de actuar como herramienta para que artistas, diseñadores visuales y miembros de otras comunidades ajenos a la programación, aprendan las bases de la misma a través de una realimentación gráfica inmediata y visual de los resultados obtenidos de su experiencia de programación.

El lenguaje de Processing se basa en Java, aunque hace uso de una sintaxis simplificada y de una biblioteca sencilla para generación de gráficos.

Ilustración - Logo de Processing

Más adelante podemos apreciar un extracto de código de Processing viendo la similitud con el código Arduino. Al correr este ejemplo podemos observar como renderiza visualmente el código en el visor (Ilustración 5 - Processing ejemplo)

void setup() {

size(480, 120);

}

void draw() {

if (mousePressed) {

fill(0);

} else {

fill(255);

}

ellipse(mouseX, mouseY, 80, 80);

}



Ilustración - Processing ejemplo

### 3.2.3 Fritzing

El entorno de software Fritzing ayuda a los diseñadores y artistas a documentar sus prototipos interactivos y dar paso en la creación de prototipos físicos al producto real. Como podemos apreciar en la siguiente ilustración (Ilustración 6 - Entorno Fritzing), permite arrastrar componentes y generar un sketch. Fritzing es creado bajo los principios de Processing y Arduino, y permite a los usuarios a documentar sus prototipos basados en Arduino y crear esquemas de circuitos impresos para su posterior fabricación

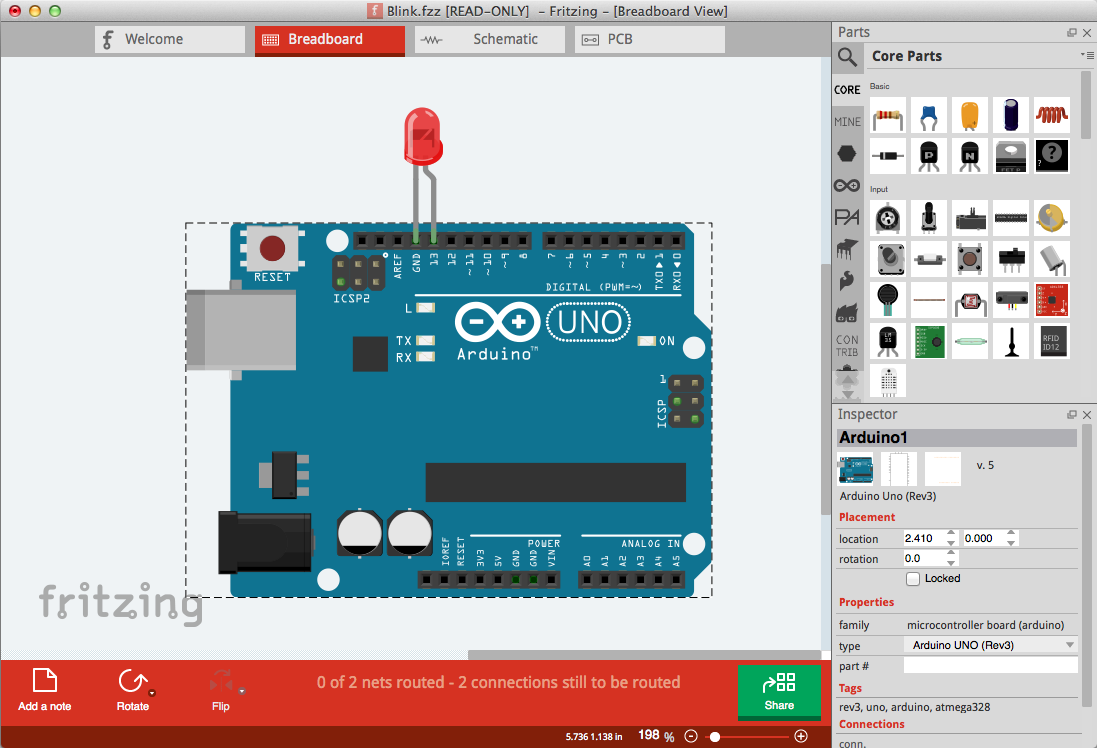


Ilustración - Entorno Fritzing

## **3.3 Características generales de la plataforma**

* Arduino es una plataforma de hardware libre, basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinares.
* Arduino es una plataforma de hardware abierto que facilita la programación de un microcontrolador. Los microcontroladores nos rodean en nuestra vida diaria, usan los sensores para ecuchar el mundo físico y los actuadores para interactuar con el mismo. Los microcontroladores leen sobre los sensores y escriben sobre los actuadores.

La plataforma consiste en una placa de circuito impreso con un microcontrolador, usualmente Atmel AVR, puertos digitales y analógicos de entrada/salida los cuales pueden conectarse a placas de expansión (*shields*), que amplían las características de funcionamiento de la placa Arduino. Asimismo, posee un puerto de conexión USB desde donde se puede alimentar la placa y establecer comunicación con el computador.

Las placas Arduino además incluyen puertos serie, uno de ellos asociado a la conexión USB a la computadora a través de una UART (Glosario) .

Por otro lado, también opera en nivel TTL (transistor-transistor logic). Esto significa que la comunicación se realiza mediante variaciones en la señal entre 0V y Vcc (donde Vcc suele ser 3.3V o 5V). Por el contrario, otros sistemas de transmisión emplean variaciones de voltaje de -Vcc a +Vcc (por ejemplo, los puertos RS-232 típicamente varían entre -13V a 13V).

Como podemos observar en la siguiente ilustración (Ilustración 7 - Ejemplo serie), se realiza una comunicación serie a (9600 bps) imprimiendo un contador. La zona marcada con rojo, es un botón que al presionarlo nos permite acceder a la terminal y ver el flujo serie seteando el clock correspondiente.

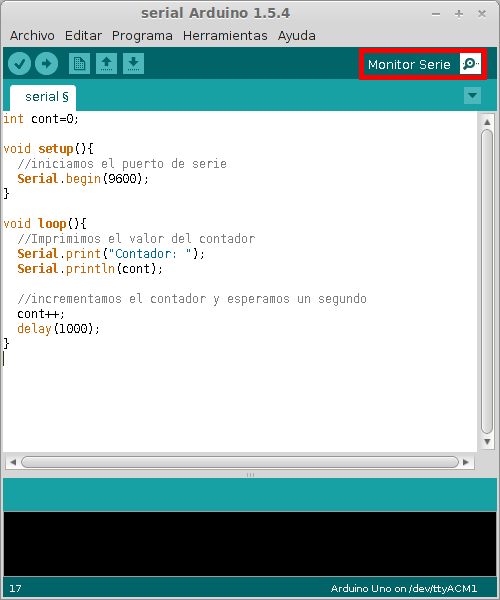


Ilustración - Ejemplo serie

3.4 Distintas plataformas para Arduino

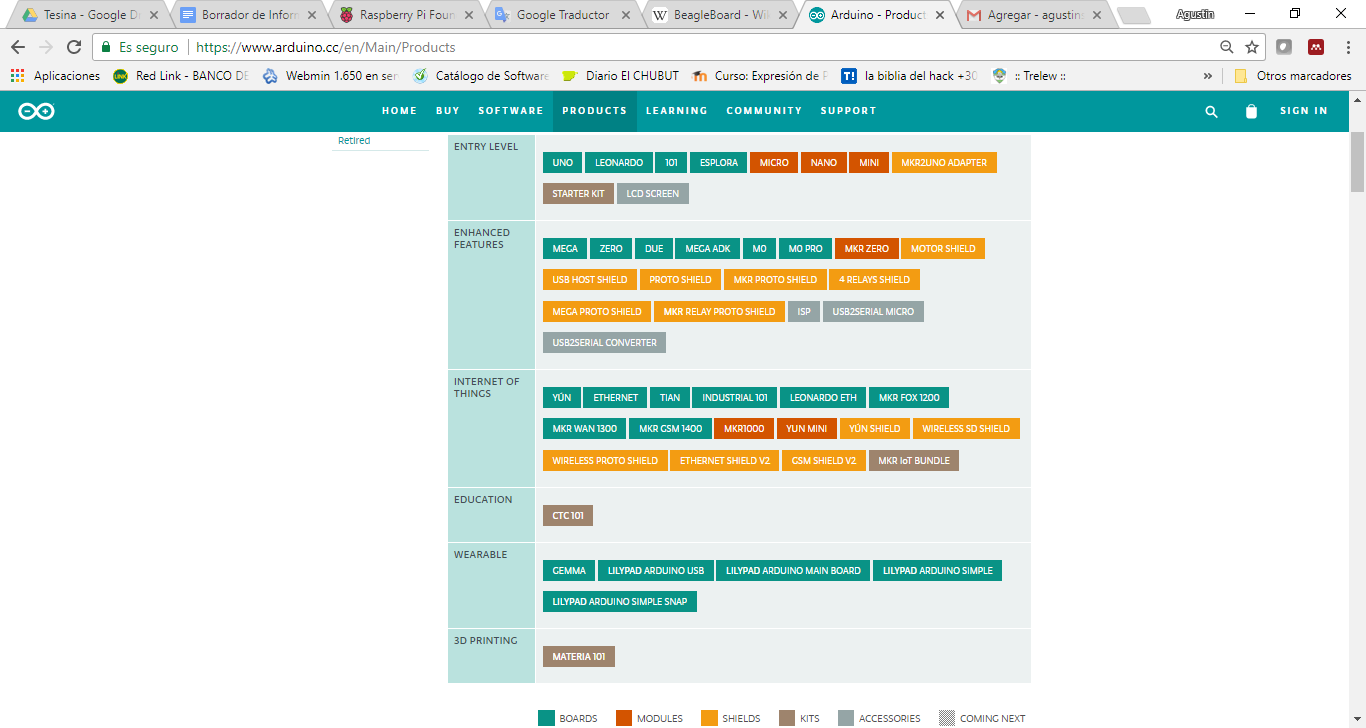


Ilustración - Niveles de entrada a la plataforma Arduino

Existe una gran variedad de productos Arduino, la compañía los cataloga, como se puede aprecias en (Ilustración 8 - Niveles de entrada a la plataforma Arduino), en distintos niveles según su utilidad[[2]](#endnote-3):

* Nivel de entrada: Son los más sencillos de utilizar, ideales para comenzar con la plataforma Arduino y realizar proyectos sencillos.
* Características mejoradas: Estas plataformas poseen características superiores, con respecto a las del nivel de entrada, están pensadas para proyectos más avanzados o de respuesta más rápida.
* Internet de las cosas: Estas placas vienen incorporadas con componentes que permitan realizar trabajos relacionados con la IoT (Internet de las cosas Glosario) mediante la incorporación de hardware de conectividad.
* Educación: En este caso, Arduino, ofrece un kit con herramientas y más de 25 proyectos, orientados a la educación, para realizar con sus plataformas.
* Usables: Estas plataformas están pensadas para “agregarle algo de electrónica” a prendas de vestir.
* Impresión 3D: Arduino ofrece una impresora 3D nombrada como Materia 101.

El hardware Arduino más sencillo consiste en una placa con un microcontrolador y una serie de puertos de entrada y salida. Los microcontroladores de 8 bits de AVR más utilizados en estas placas son el Atmega168, Atmega328, Atmega1280, y Atmega8 por su sencillez y bajo coste, aunque también se dispone de microcontroladores ARM, cómo el caso del CortexM3 de 32 bits. A pesar de que ARM y AVR son plataformas diferentes, al utilizar la IDE de Arduino, los programas se compilan y luego se ejecutan sin cambios en cualquiera de las plataformas. En la imagen (Ilustración 9 - Arduino Uno) se visualiza la distribución física de puertos y componentes de la versión Arduino Uno R3.

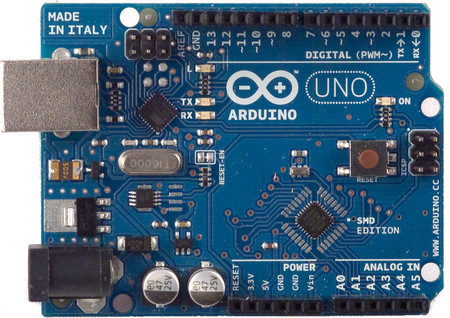


Ilustración - Arduino Uno

Una primera diferenciación entre los distintos modelos de Arduino la encontraremos en el voltaje o tensión de alimentación de las placas. Las basadas en CortexM3 operan con un voltaje de 3,3 voltios, mientras que la mayor parte de las placas basadas en AVR utilizan una tensión de 5 voltios. Esto de todas formas no es un factor decisivo en la elección de una placa, dado que existen conmutadores de tensión en muchos actuadores y sensores compatibles.

## **3.5 Aplicaciones**

Los usos posibles que se le pueden dar a “un Arduino”, en forma general son:

* Utilizarlo como microcontrolador, con un programa descargado desde un ordenador y funcionamiento de forma independiente, recibiendo entradas de sensores y realizando acciones sobre actuadores en función de las entradas y el programa.
* Idem anterior pero conectado a un ordenador (que también podría ser un SBC como Raspberry Pi).

Enumerarenmos ahora una serie de razones de utilización

## **3.6 Motivaciones para su uso**

### 3.6.1 La comunidad

Arduino cuenta con una gran comunidad, cuayas actividades se centran en la experimentación, publicación de resultados y proyectos, y orgnzación de eventos.

evangelizador

Dentro de la página oficial se brinda soporte por medio de documentación, foros y la publicación de un blog con novedades y proyectos relevantes que se encuentran en desarrollo.

Además, se han creado sitios como Arduino Playground (<http://playground.arduino.cc/>) que consiste en una Wiki donde todos los usuarios de Arduino pueden contribuir. Es el lugar donde publicar y compartir código, diagramas de circuitos, guías, manuales, cursos. Es una de las base de datos de conocimiento de la comunidad de Arduino. Este sitio a su vez tiene soporte de distintos lenguajes como el español (<https://playground.arduino.cc/Es/Es>).

Otro ejemplo de las actividades de la comunidad es el sitio Arduino Hub, un lugar donde se comparten los proyectos, dando los distintos pasos para reproducirlo.

El Arduino day, o cumpleaños de Arduino, es una celebración mundial que se lleva a cabo una vez al año en diversos puntos del mundo. Este evento es organizado por la comunidad de Arduino y/o sus fundadores. En él se desarrollan diferentes talleres, charlas y concursos, entre otras actividades, relacionadas con la plataforma.

La siguiente imagen (Ilustración 10 - Logotipo comunidad open-source de Arduino) muestra el logotipo oficial de la comunidad open-source de Arduino.



Ilustración - Logotipo comunidad open-source de Arduino

### 

### 3.6.2 Sencillez de programación

Gracias a la reutilización de las ideas de Wiring, Arduino provee un alto nivel de abstracción con respecto al hardware.

Por ejemplo, para establecer como salida los los puertos 1 al 7, se puede utilizar el siguiente fragmento de código:

PinMode(1,OUTPUT) ;

PinMode(2,OUTPUT) ;

…;

PinMode(7,OUTPUT) ;

en contraposiciic con:

DDRD = B11111110; [[3]](#footnote-4) // sets Arduino pins 1 to 7 as outputs, pin 0 as input

### 3.6.3 Hardware económico

Lo único que “vale” en la placa son sus componentes, ya que no debemos pagar el costo de la licencia de su creador, por el hecho de ser hardware libre.

## **3.7 Incorporación de Arduino en las escuelas**

Las deviersas características y motivaciones hacen atractiva a la plataforma Arduino para su incorporación en las escuelas.

Esto ha llevado la c.

Dentro de las principales características que han promovido esta tendencia sen encuentran la sencillez del lenguaje de programación que permite que alumnos y docentes no necesariamente del ámbito de la informática y la electrónica pueden utilizarlo, contribuyendo a la construcción colectiva del conocimiento, promoviendo la interdisciplinariedad escolar, permitiendo la colaboración de docentes de distintas áreas cooperar en la articulación de proyectos.

Desde el punto de vista pedagógico del proceso de aprendizaje , este tipo de actividades permiten al sujeto que aprende ser participante activo, desde la concepción de la idea hasta el producto final, incorporando gradualmente conocimientos técnicos específicos.

Este tipo de actividades educativas hacen que la tecnología y su uso se pongan al servicio de la creatividad, el juego, la experimentación y la invención, con la posibilidad de ser adaptado al contexto en el que se inserta. Además, proporcionar la recuperación de la tecnología obsoleta existente en ellas como se describe en la siguiente sección.

### 3.7.1 Las tres erres

Las tres erres (reducir, reutilizar, reciclar) es una regla para cuidar el medio ambiente, específicamente para reducir el volumen de residuos o basura generada.

Cuando hablamos de reducir lo que estamos diciendo es que se debe tratar de simplificar el consumo de los productos directos.

Al decir **reutilizar**, nos estamos refiriendo a poder volver a utilizar los objetos y darles la mayor utilidad posible antes de que llegue el momento de desecharlos.

Por otro lado, **reciclar** consiste en el proceso de someter los materiales a una transformación en el cual se puedan volver a utilizar.

Esta definición se pretende aplicar en las escuelas haciendo un proceso de clasificación, selección y desoldando componentes electrónicos de placas en desuso y materiales que se han desechado en las instituciones o en hogares de los alumnos.

## **3.8 Actuadores y sensores**

Un **actuador** es un dispositivo capaz de transformar energía hidráulica, neumática o eléctrica en la activación de una acción con la finalidad de generar un efecto sobre un proceso automatizado. Este recibe la orden de un regulador o controlador y en función a ella genera la orden para activar un elemento final de control, como por ejemplo un LED.

Por otro lado, un **sensor** es un objeto capaz de detectar magnitudes físicas o químicas, llamadas variables de instrumentación, y transformarlas en variables eléctricas. Las variables de instrumentación pueden ser, por ejemplo: intensidad lumínica, temperatura, distancia, aceleración, inclinación, presión, desplazamiento, fuerza, torsión, humedad, movimiento, pH, etc.

En conjunto, los sensores y actuadores, permiten la creación de distintos tipos de artefactos, que posibilitan comunicarse con el ambiente que los rodea, modificándolo (actuadores) o recibir estímulos (sensores).

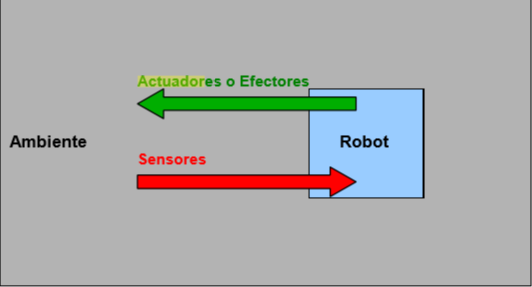


Ilustración - Representación actuadores y sensores

En esta imagen (Ilustración 11- Representación actuadores y sensores) se representan los datos que un robot puede capturar de su ambiente por medio de diversos sensores, y a su vez como podría interactuar con el mismo mediante actuadores.

## **3.9 Actuadores en el SAR**

La electrónica industrial ha generado estandarización en el campo de los sensores y actuadores, muchos de estos últimos con buen soporte en Arduino. Precisamente en el SAR se utilizarán:

* Motores de corriente continua
  + Para el desplazamiento del robot móvil
* LED
  + Para indicar estados del RM

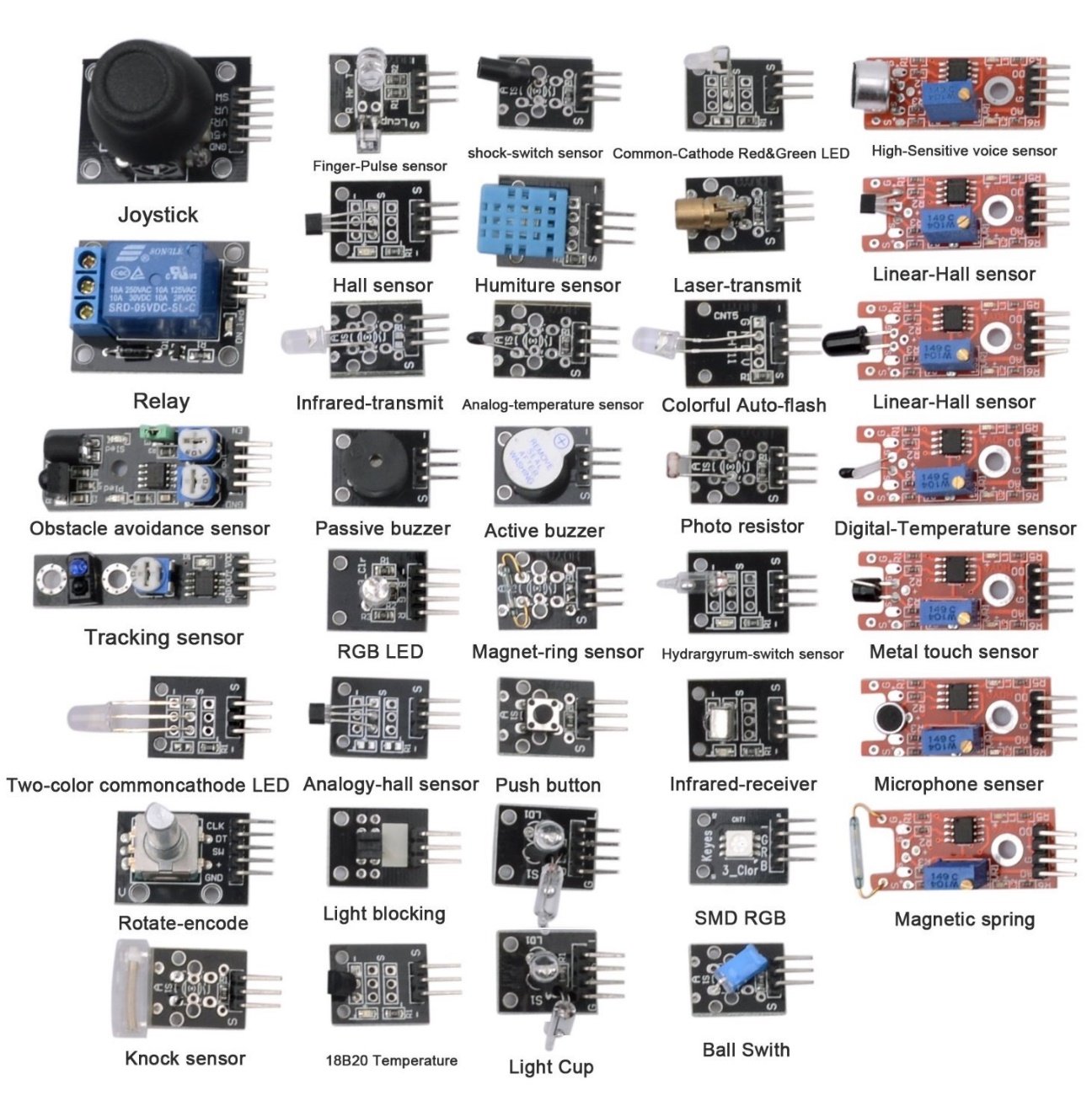


Ilustración - Actuadores y sensores compatibles con Arduino

## **3.10 Sensores en el SAR**

El SAR utiliza los siguientes sensores:

* Sensor ultrasónico HC-SR04
  + Para detectar objetos, y distancia entre el RM y elementos del ambiente
* Sensor de Temperatura KY-001
  + Incorporado para analizar la temperatura del ambiente
* Sensor de presencia de gases MQ-7
  + Detección de monóxido de carbono

Algunos de los sensores y actuadores se pueden apreciar en la ilustración anterior (Ilustración 12 - Actuadores y sensores compatibles con Arduino).

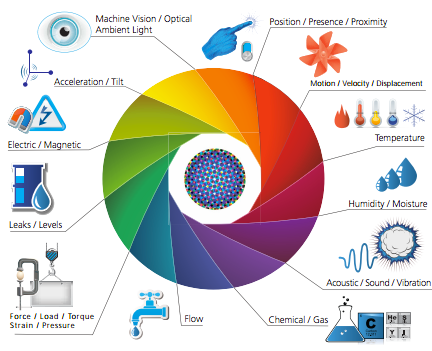


Ilustración - Representación de sensores

En esta imagen (Ilustración 13- Representación de sensores) se pueden apreciar los distintos factores de un entorno que pueden ser evaluadores con sensores mencionados anteriormente.

## **3.11 Módulos o *shields* en el SAR**

El SAR utiliza[[4]](#footnote-5):

* MotorShield L298
  + Para administración del puente H y gestión de los motores de CC
* Módulo bluetooth HC-05
  + Para la comunicación con dispositivos compatibles (móviles y/o computadoras)
  + Envío de órdenes
* Módulo GPS NEO-6
  + Para la geolocalización del RM
* Módulo ESP8266
  + Conectividad y transferencia de datos vía WIFI
  + Activación del modo AP

A lo largo del desarrollo de la tesina se fueron implementando diversos casos de pruebas sobre los sensores, actuadores y módulos especificados en esta sección. Las pruebas se encuentran anexas en este documento.

## **3.12 Resumen**

Como vimos en el presente capítulo, Arduino es una plataforma electrónica open-source, basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinares. A su vez facilita la programación de un microcontrolador, este último lee sobre los sensores y escribe sobre los actuadores.

Un actuador es un dispositivo capaz de transformar energía hidráulica, neumática o eléctrica en la activación de una acción con la finalidad de generar un efecto sobre un proceso automatizado.

Por otro lado, un sensor es un objeto capaz de detectar magnitudes físicas o químicas, llamadas variables de instrumentación, y transformarlas en variables eléctricas. Además, existen módulos que integran sensores y actuadores con un micro controlador

1. http://comoprogramarpic.blogspot.com.ar/2012/06/programando-un-atmel-mi-primer-programa.html [↑](#endnote-ref-1)
2. <https://www.arduino.cc/en/Main/Products> [↑](#endnote-ref-3)
3. <https://www.arduino.cc/en/Reference/PortManipulation> [↑](#footnote-ref-4)
4. El uso de algunos de estos módulos queda en forma tentativa, dado que existen también en la Raspberry y su uso puede ser complementario. [↑](#footnote-ref-5)