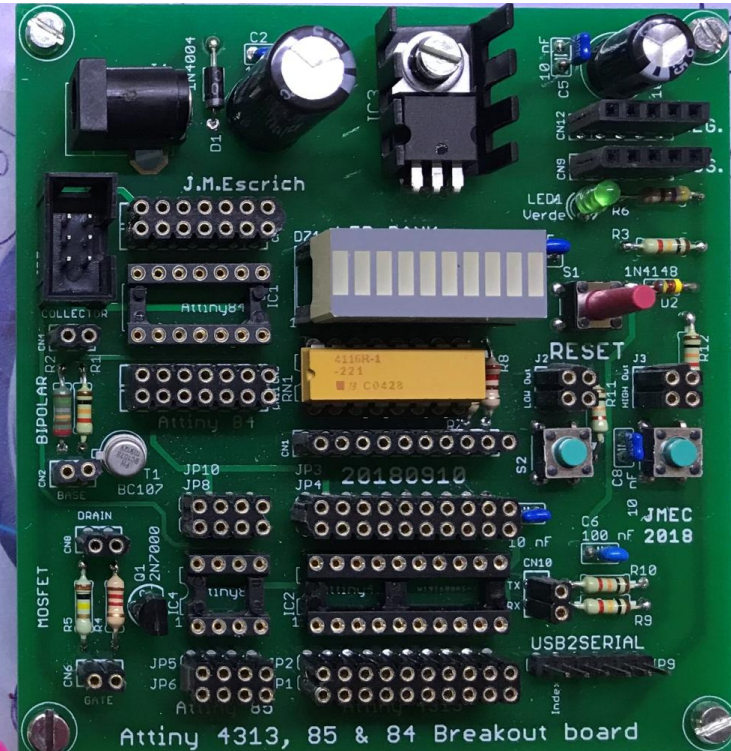
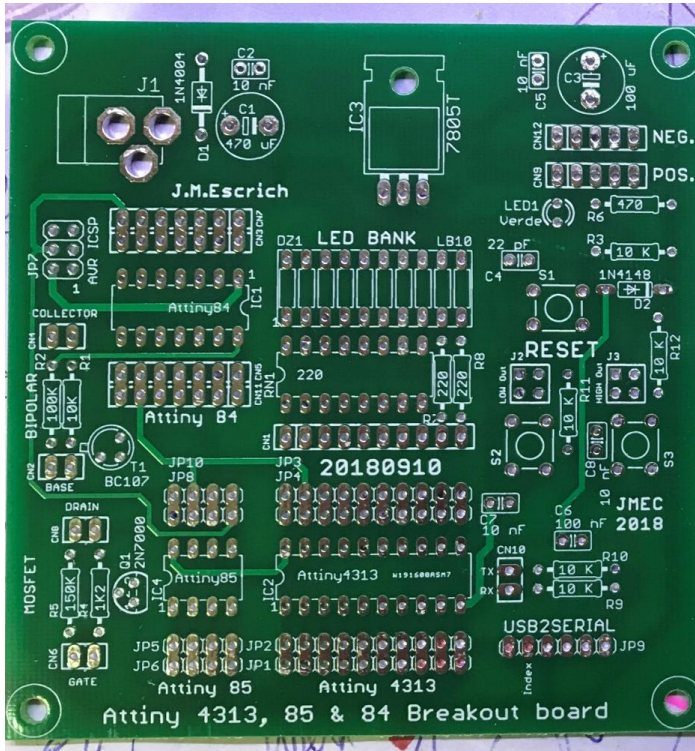


Electrónica digital

(de andar por casa)



Prefacio:

Hola a todos, como siempre andáis con problemas preguntándoos que resistencia tengo que ponerle a un led o como medir con vuestro Arduino la tensión de un batería, os he preparado este pequeño manual, que se basa fundamentalmente en los diferentes modelos de Arduino y dispositivos compatibles, aunque la mayoría de lo que aquí exponga es valido para cualquier otro tipo de microcontrolador o procesador, así como para sistemas, montajes o circuitos, que no lleven ningún elemento programable, voy a tratar de explicarlo de manera simple y de usar las matemáticas lo menos posible, a ver si consigo transmitir lo que quiero contaros.

En caso de errores, dudas, preguntas, mi correo electrónico, que todos tenéis, es:

xxxxxxx@xxxx.com

En primer lugar vamos a ver que es la electrónica digital y su diferencia con la electrónica analógica.

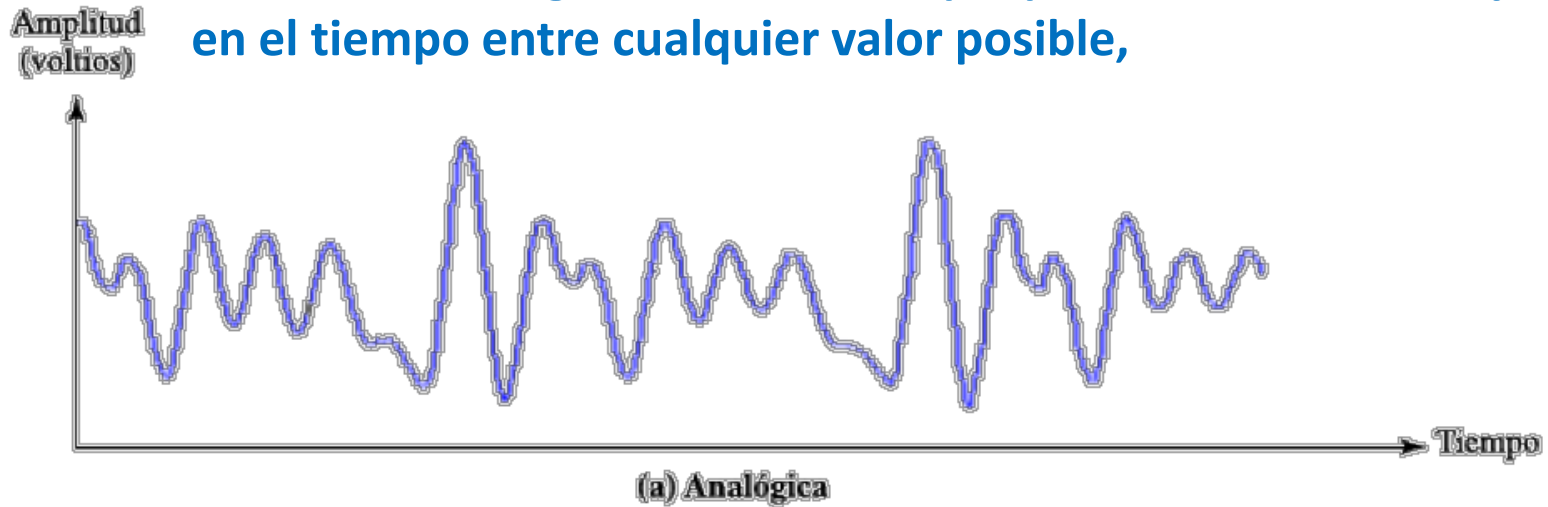
En la electrónica analógica nos importa su valor en voltios, si varia o no varia, si es corriente continua o alterna, la frecuencia y algunas otras cosas mas dependiendo de cada aplicación.

En la electrónica digital no nos importa tanto el nivel de tensión, si no el hecho de que haya o no haya tensión, esa tensión y el hecho de que esté o no esté es lo que define en principio a la electrónica digital.

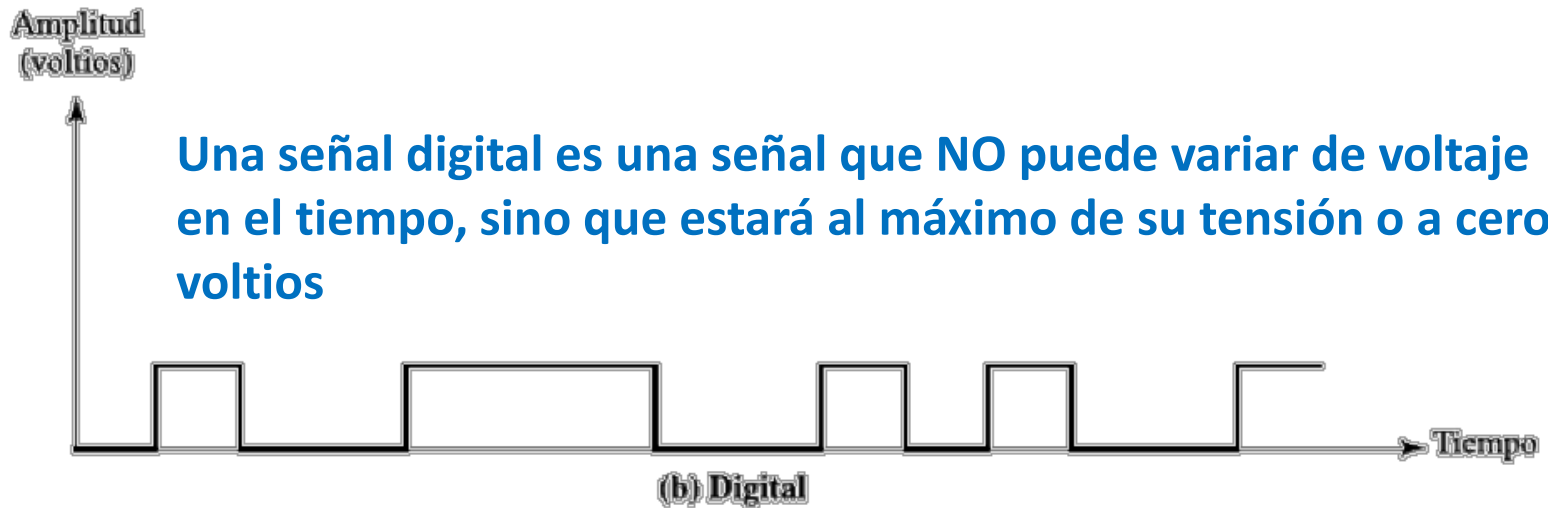
A continuación unos ejemplos.

Señales analógicas y digitales

Una señal analógica es una señal que puede variar de voltaje en el tiempo entre cualquier valor posible,



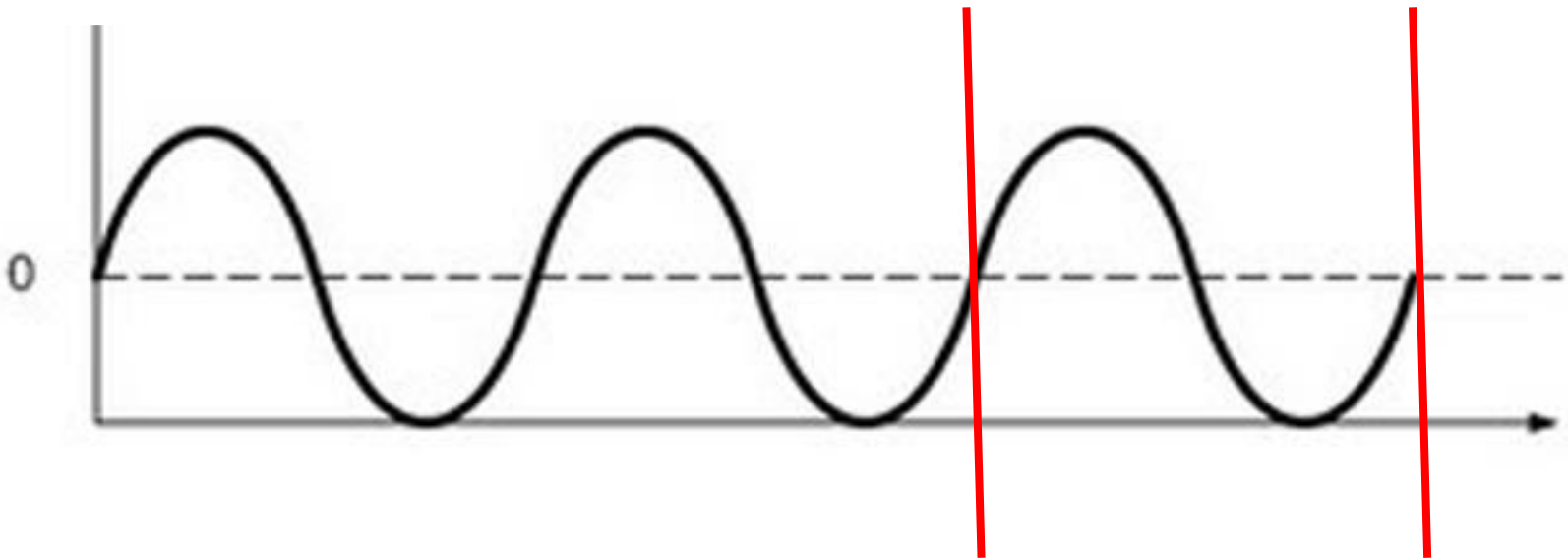
Una señal digital es una señal que NO puede variar de voltaje en el tiempo, sino que estará al máximo de su tensión o a cero voltios



Señales analógicas y digitales

Esta es la forma que hace la tensión que hay en el enchufe de vuestra casa, la parte que está entre las dos líneas rojas se repite 50 veces por segundo, esa es la frecuencia de 50 Hercios que tiene la corriente alterna en vuestra casa, veréis que la mitad del tiempo la onda es positiva y la mitad del tiempo es negativa, que alterna entre ambos estados, por eso es corriente alterna, cada segundo pasa 100 veces por cero voltios, y aunque vosotros veis 220 voltios realmente ese es lo que se llama el valor eficaz, la tensión máxima o tensión de pico sobrepasa los 300 voltios, se calcula así:

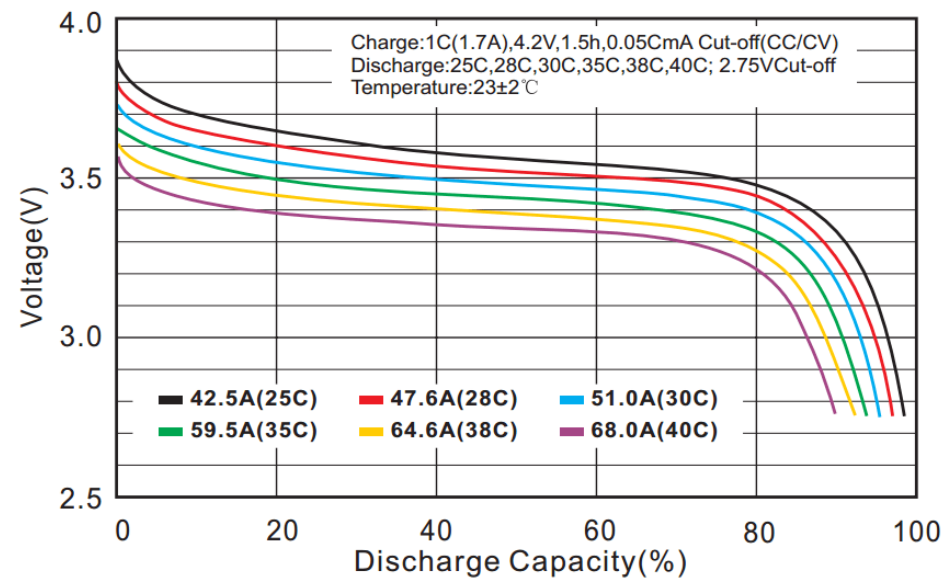
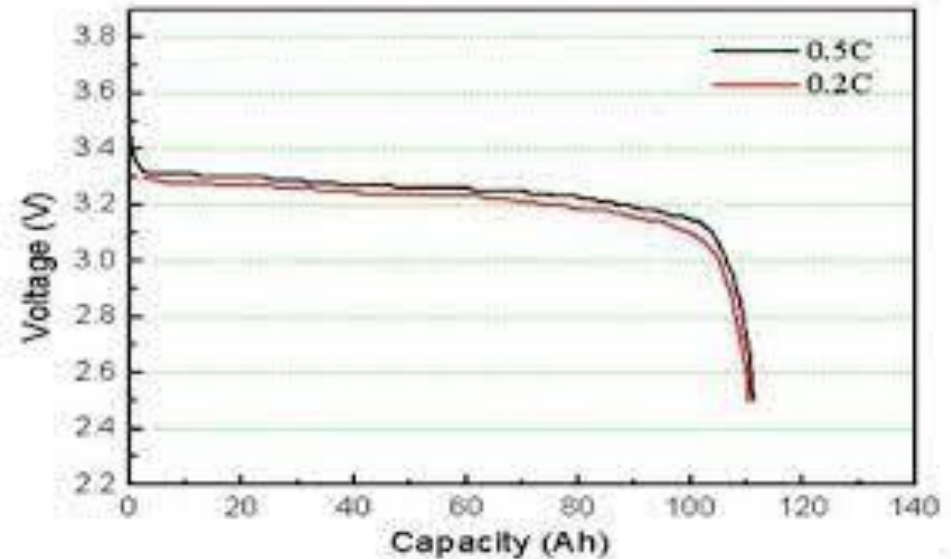
$$220 \times (\text{Raíz cuadrada de } 2) = 311,27 \text{ voltios}$$



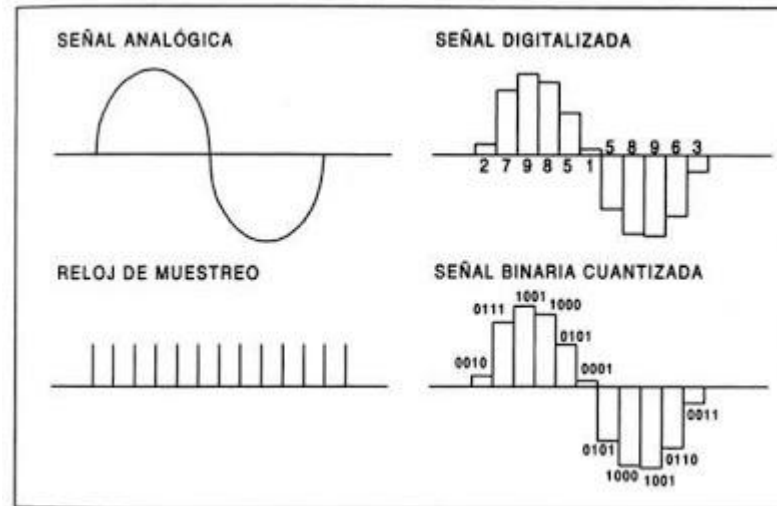
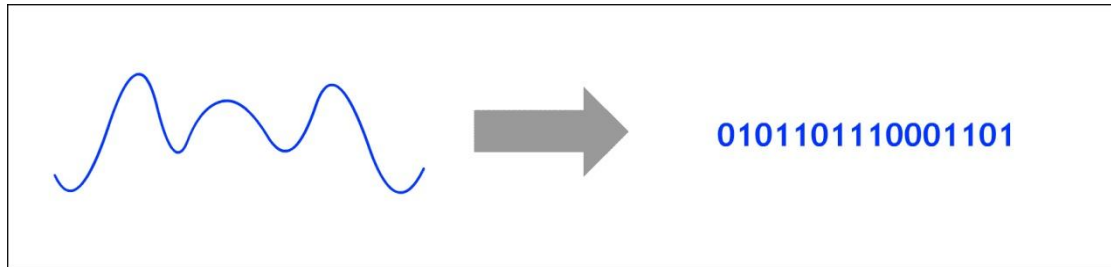
Señales analógicas y digitales

Pero una señal analógica no tiene porque ser siempre una corriente alterna, también puede ser una corriente continua.

En la practica, analógico es todo lo que no sea digital, aquí tenéis un grupo de señales analógicas que seguro que conocéis y que se aproximan mas a lo que os interesa, las curvas de tensión de una batería en función de la carga que tienen.



Señales analógicas y digitales

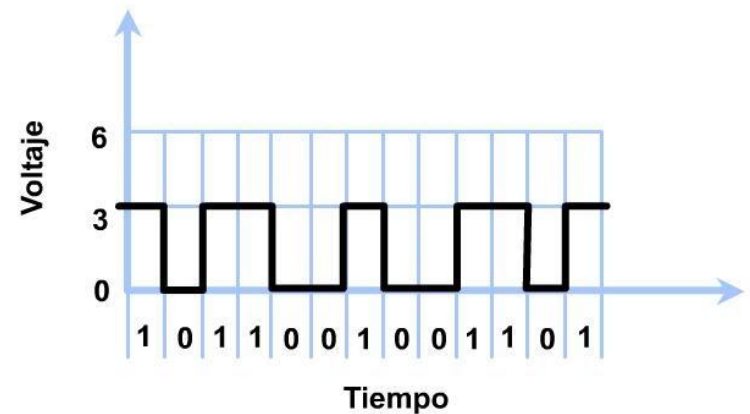


Está claro que lo que nos interesa a nosotros, es poder convertir una señal analógica en su equivalente digital

Tratamiento de señales digitales

Para poder operar con una señal digital, necesitamos tener algunas cosas en cuenta, el nivel bajo o cero, se reconoce como tal por debajo de unos 0,8 voltios, el nivel alto o uno, se reconoce como tal por encima de 2 voltios, esto es en un circuito integrado típico como por ejemplo el SN7400, en diferentes circuitos integrados estos niveles varían ligeramente, en cualquier caso, lo que nos queda claro, es que tenemos los dos niveles definidos y un área de incertidumbre en el centro que tenemos que evitar, y que mas adelante veremos como, para las entradas digitales.

Señal digital



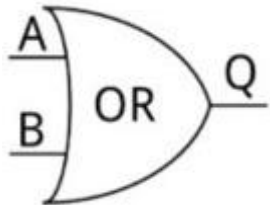
Valores digitales

Para que vayáis cogiendo el concepto, os dejo esta tabla con ejemplos de alto y bajo, es simple y se entiende bien, espero

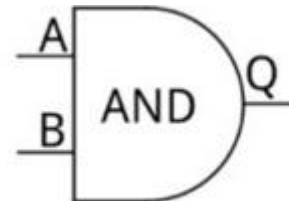
Desconectado	Conectado
0	1
LOW	HIGH
0 Voltios	5 Voltios
Bajo	Alto
Apagado	Encendido

Puertas lógicas

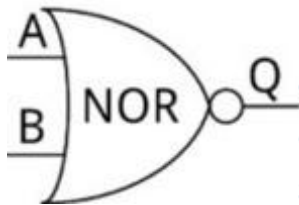
La electrónica digital está basada en estos cinco circuitos que os muestro a continuación, estas son las formas básicas, cualquier otro dispositivo digital, incluso el ordenador mas complejo, está hecho con combinaciones de estos circuitos básicos, esta es la descripción de lo que hacen:



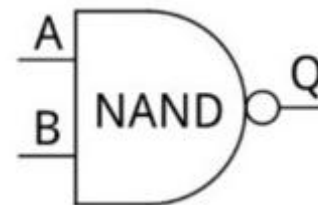
Puerta OR, cuando cualquiera de sus entradas es un uno, la salida será un uno



Puerta AND, cuando todas sus entradas están puestas a uno, la salida será un uno



Puerta NOR, cuando cualquiera de sus entradas es un uno, la salida será un cero



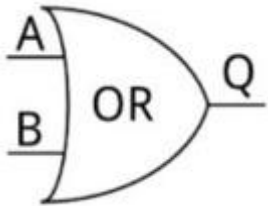
Puerta NAND, cuando todas sus entradas están puestas a uno, la salida será un cero



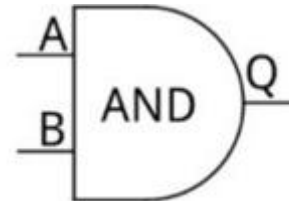
Puerta NOT, o inversor, cuando su entrada está a uno, la salida será un cero y viceversa

Puertas lógicas

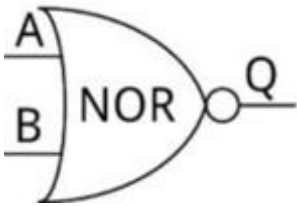
El comportamiento de estas puertas, según la descripción anterior, se muestra en unas tablas que se llaman tablas de la verdad, y que muestran todas las combinaciones posibles, estas son las tablas de la verdad, observad que la bolita siempre significa la negación.



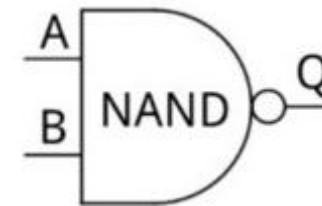
Puerta OR		
A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1



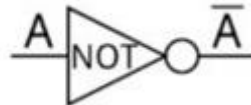
Puerta AND		
A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1



Puerta NOR		
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0



Puerta NAND		
A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0



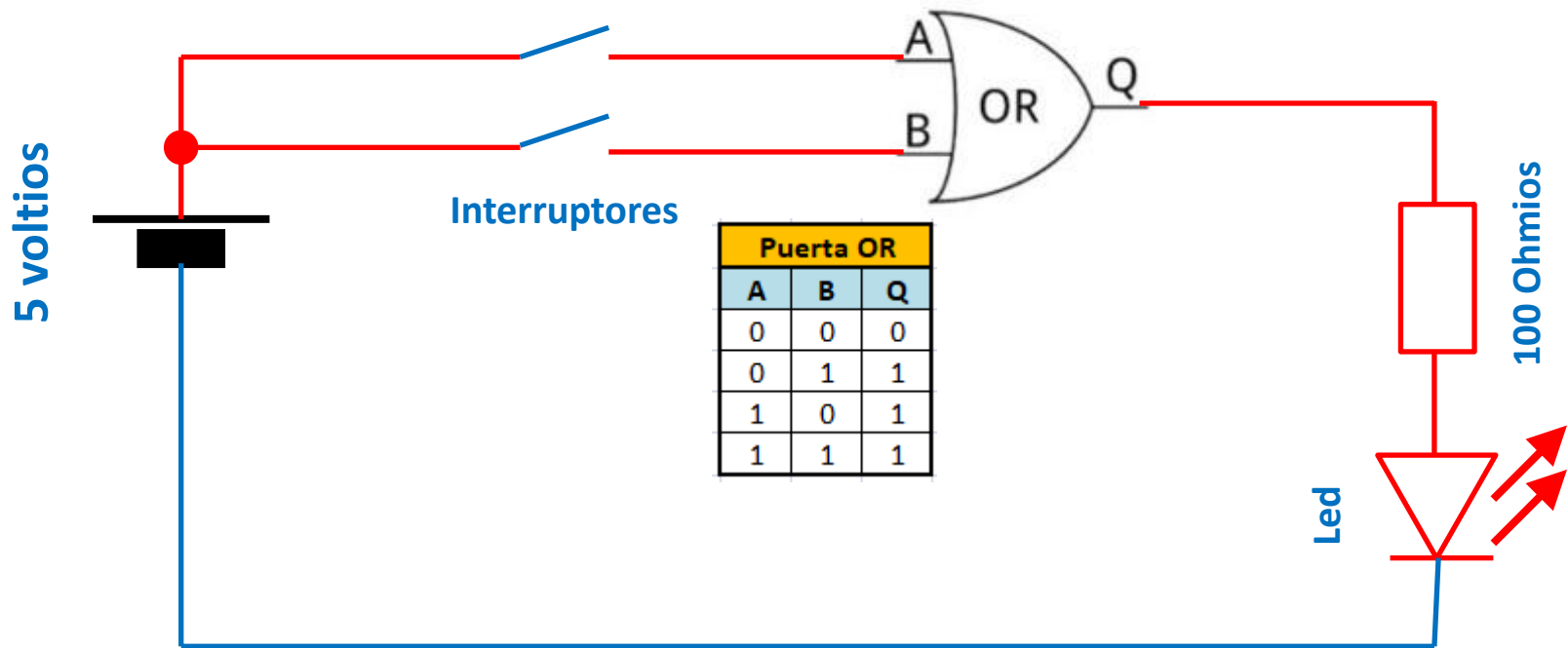
Puerta Not		
A		Q
0		1
1		0

Aunque todas las puertas menos la última tienen dos entradas, es posible encontrarlas en el mercado con hasta ocho entradas y en algunos casos incluso diez.

Operando con puertas lógicas

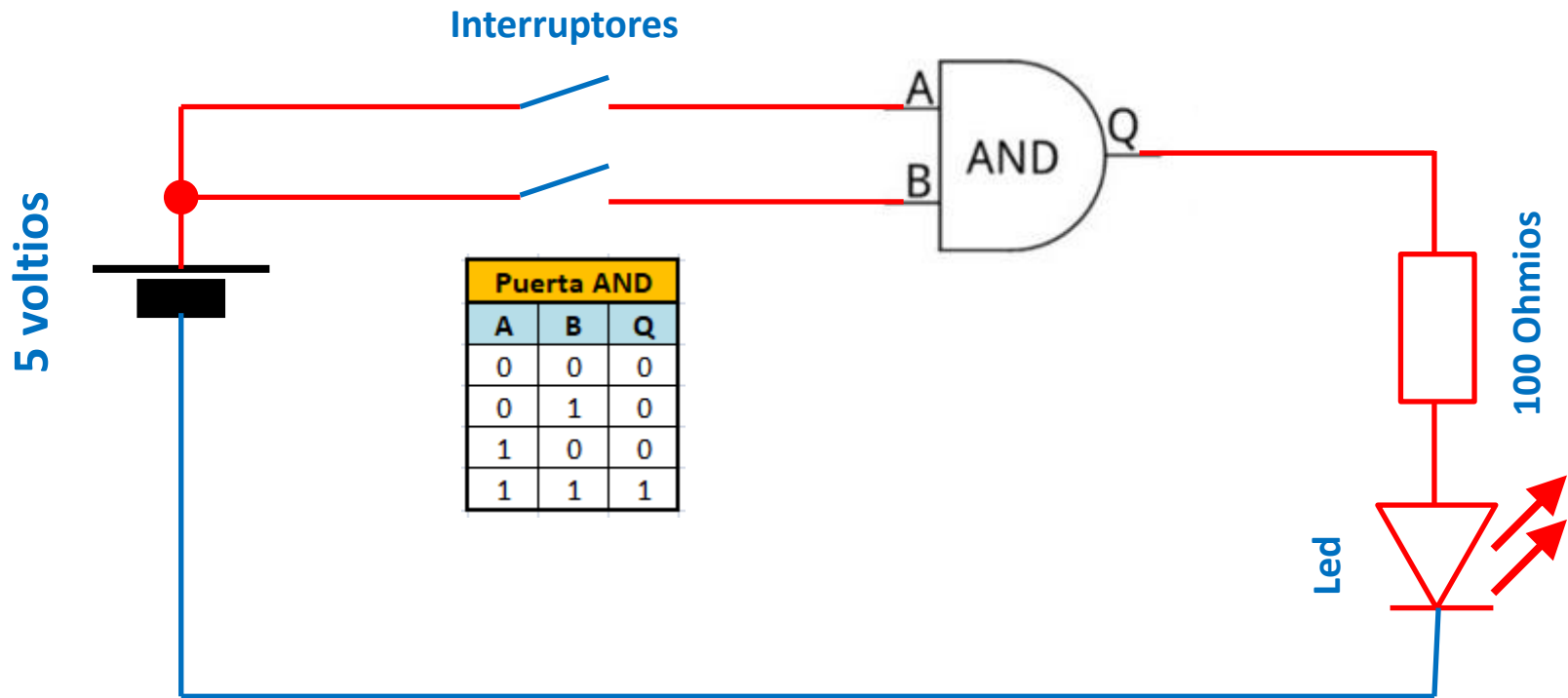
Me imagino que a estas alturas estaréis diciendo, no, pero si todo esto está muy bien, ¿pero que para que me sirve a mi?, ¿que como uso yo esto?, pues bien, vamos a usar una puerta AND, para hacer unos experimentos, recordad que está alimentada a cinco voltios y que las entradas y salidas son señales de cinco voltios, que nuestros ceros son cero voltios, y nuestros unos son cinco voltios.

Bueno en este esquema usamos dos interruptores para encender un led, si conectas uno cualquiera de los dos, el led enciende, según la tabla de la verdad tanto si conectas A, como si conectas B, o los dos a la vez, el led enciende, ¿lo veis?, como veis con los interruptores aplicamos toda la tensión de la batería, que en este caso es de cinco voltios



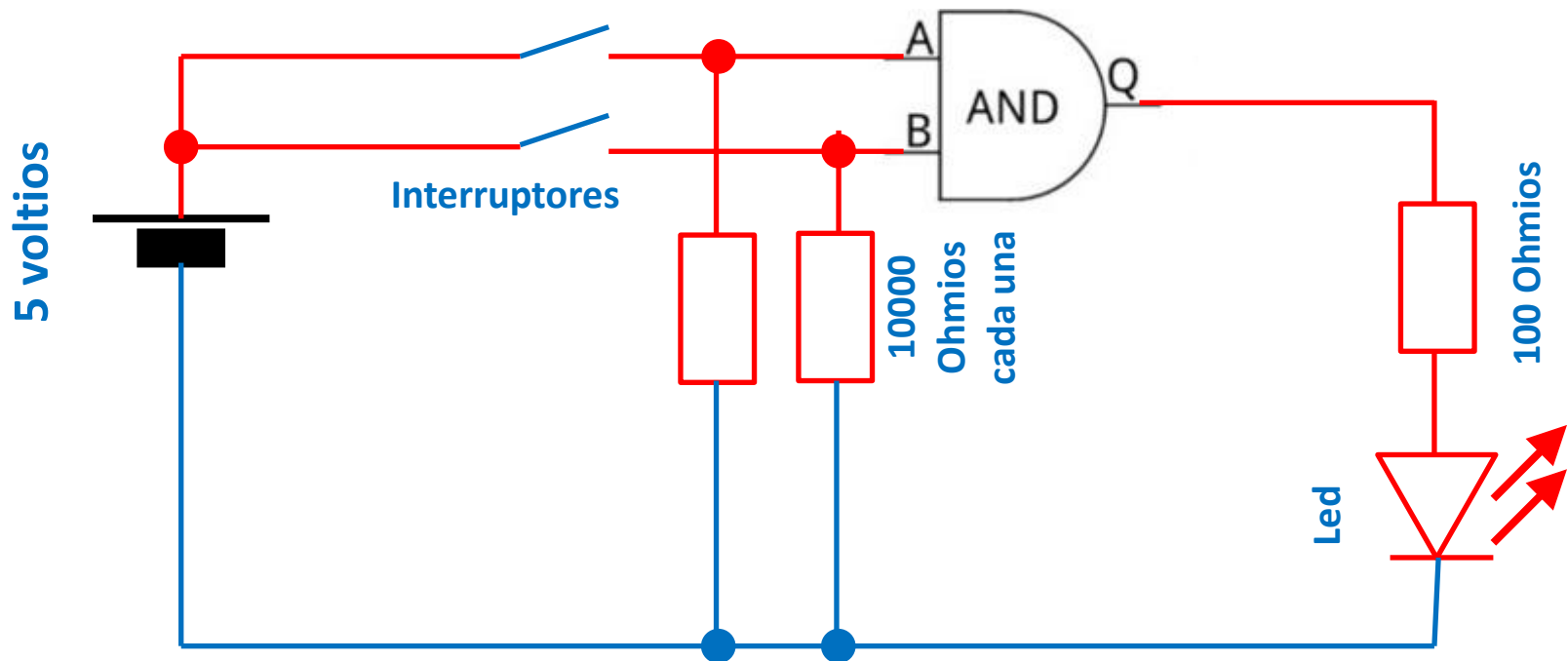
Operando con puertas lógicas

En este circuito, con los dos mismos interruptores, tenemos que tener ambos encendidos para que el led se encienda, comprobad en la tabla de la verdad que es así



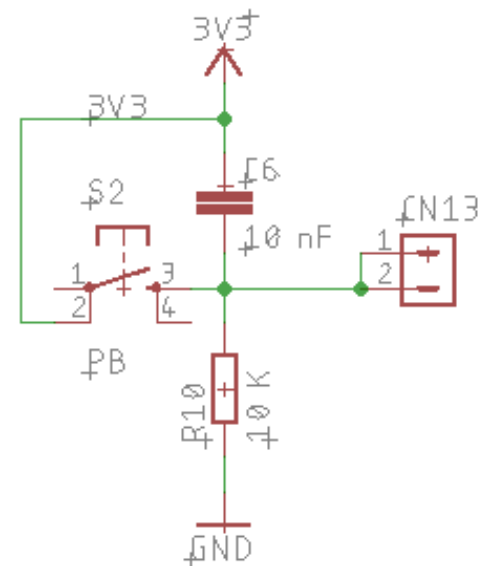
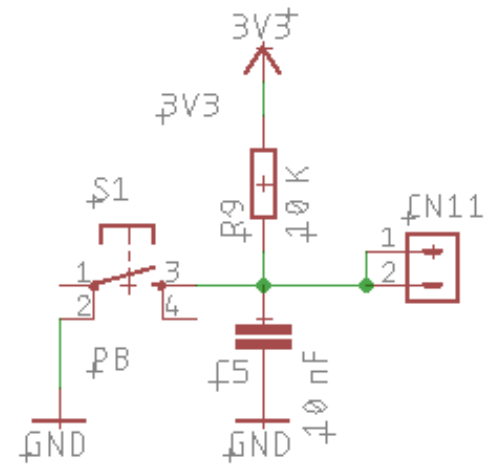
Operando con entradas digitales

Realmente ambos circuitos, que teóricamente y en clase son correctos, cuando se llevan a la practica tienen un problema, el problema es que cuando dejamos las entradas al aire, nada nos garantiza que en ellas tengamos un cero o un uno, si las dejamos al aire lo normal es que asumieran el valor 1, con lo que el led siempre estaría encendido, para eso se ponen resistencias entre la entrada y el positivo, o la entrada y el negativo, ahora funcionaría sin problemas, cuando el interruptor está desconectado la resistencia pone la entrada a cero voltios, cuando conecta se pone la entrada a cinco voltios, ¿os acordáis de la zona de incertidumbre que os decía?, pues la evitamos con estas resistencias, cuando el extremo contrario de la resistencia se conecta al positivo se llama “PULLUP”, cuando se conecta al negativo “PULLDOWN”



Operando con entradas digitales

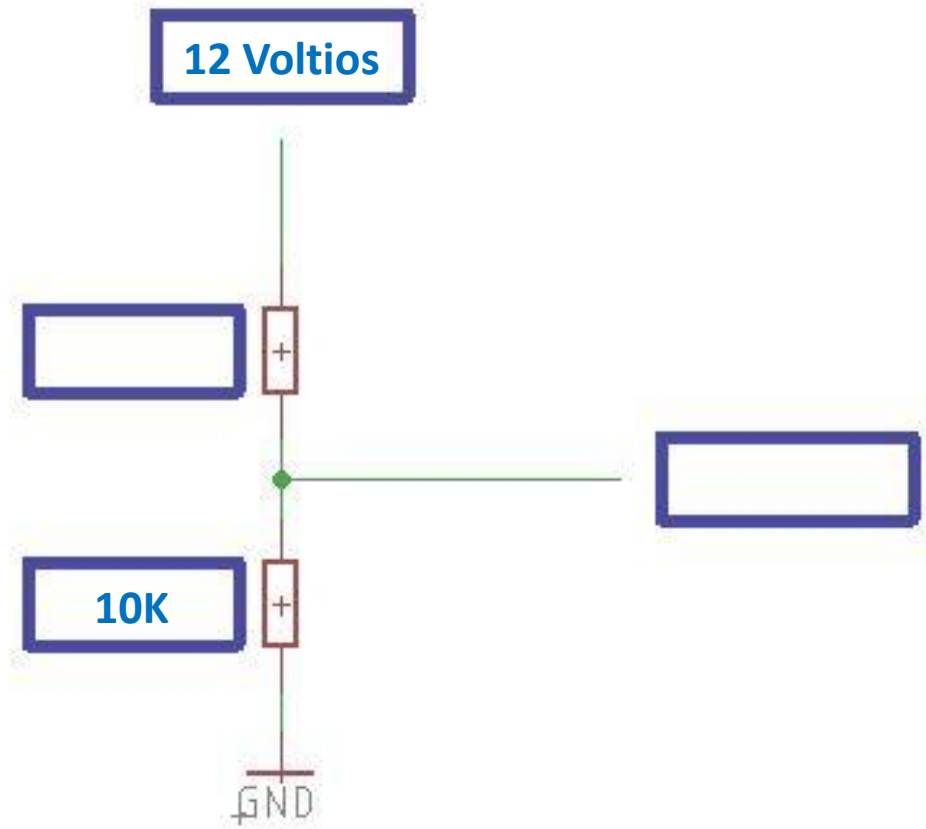
“Si bueno, pero yo he venido aquí a hablar de mi Arduino”, bien, de ello estamos hablando, las patillas digitales de un Arduino no son otra cosa que puertas lógicas, usamos puertas lógicas como entrada, o la salida de una puerta lógica como salida digital, porque nuestro programa hace que unos transistores internos conecten las puertas de entrada o las puertas de salida a las patillas del Arduino, si habéis leído la documentación, veréis que las entradas también se pueden definir como PULLUP o PULLDOWN, si no activáis estas resistencias internas para vuestras entradas, recordad poner resistencias externas para hacer esta función, la imagen de la derecha, representa parte de las entradas de otro tipo de procesador, que funciona a 3,3V, en ambos esquemas podéis ver en el de arriba un PULLUP y en el de abajo un PULLDOWN, olvidaos de momento del condensador que aparece



Divisores de tensión

Si, esto está bien, pero es una faena, ¿que pasa si yo tengo una entrada que solo aguanta cinco voltios, pero tengo una señal de 12 voltios que me viene de un termostato que tengo en mi coche que funciona a 12 voltios?, ¿tiene remedio?, claro que tiene remedio, el remedio se llama divisor de tensión.

Eso que dicho así parece tan complejo, no son mas que dos simples resistencias, como no me gusta mucho echar cuentas y para que sea fácil, la de abajo la vamos a poner de 10000 Ohmios, y según el sistema métrico internacional, vamos a abreviar diciendo que es una resistencia de 10K, ahora podemos usar el calculo de la ley de Ohm o simplemente hacer una regla de tres directa, ya que a partir de 12 voltios queremos tener solo cinco en nuestra entrada, por lo tanto tenemos que calcular la resistencia para los 7 voltios que sobran



Divisores de tensión

Entonces planteamos así nuestra ecuación, si a 5 voltios le corresponden 10000 Ohmios, a 7 voltios le corresponderán X, es decir

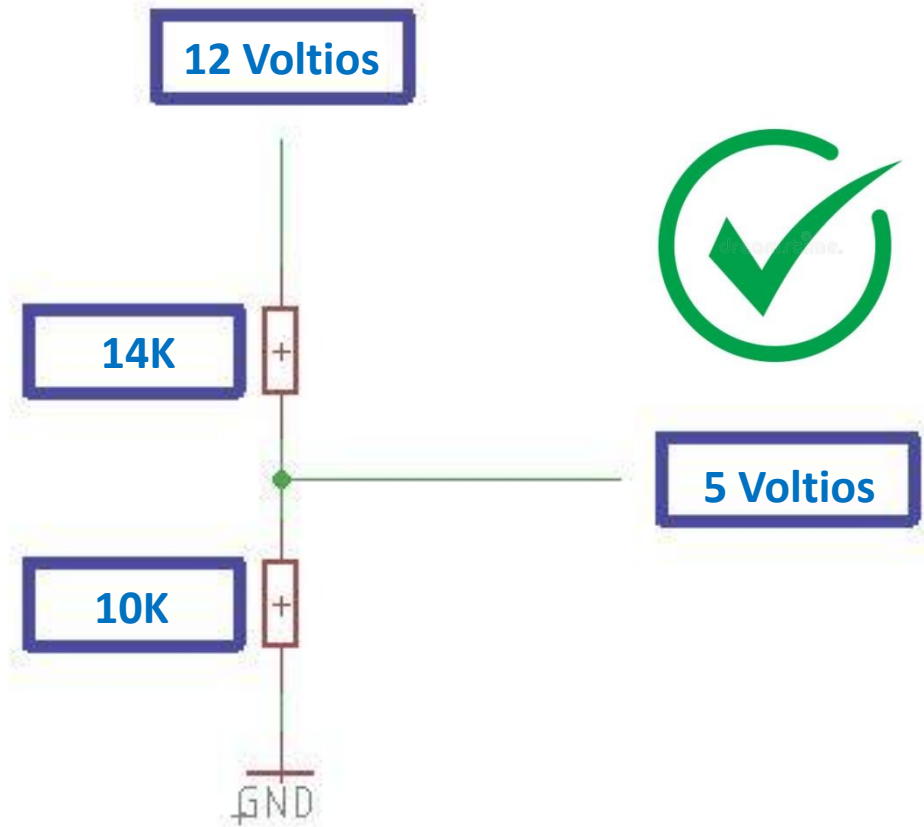
5 -----10000

7-----x

o lo que es igual, 7 multiplicado por 10000 y el resultado dividido por 5:

$$\frac{7 \times 10000}{5} = 14000$$

Por lo que si la resistencia de arriba la ponemos de 14000 Ohmios, o lo que es lo mismo, de 14K, en la salida del divisor, que va a nuestra entrada digital de nuestro Arduino, tendremos 5 Voltios, además de que nuestra resistencia de abajo nos sirve de PULLDOWN, para que la patilla no quede al aire o “Flotante”



Divisores de tensión

Si esto está perfecto, pero yo siempre tengo que venir a aguar la fiesta, resulta que las resistencias de 14K no se fabrican, ¿que faena verdad?, bueno, ¿entonces que hacemos?, pues muy sencillo, poner una aproximada, no olvidéis que a la electrónica la llaman “la ciencia del 10%”,.

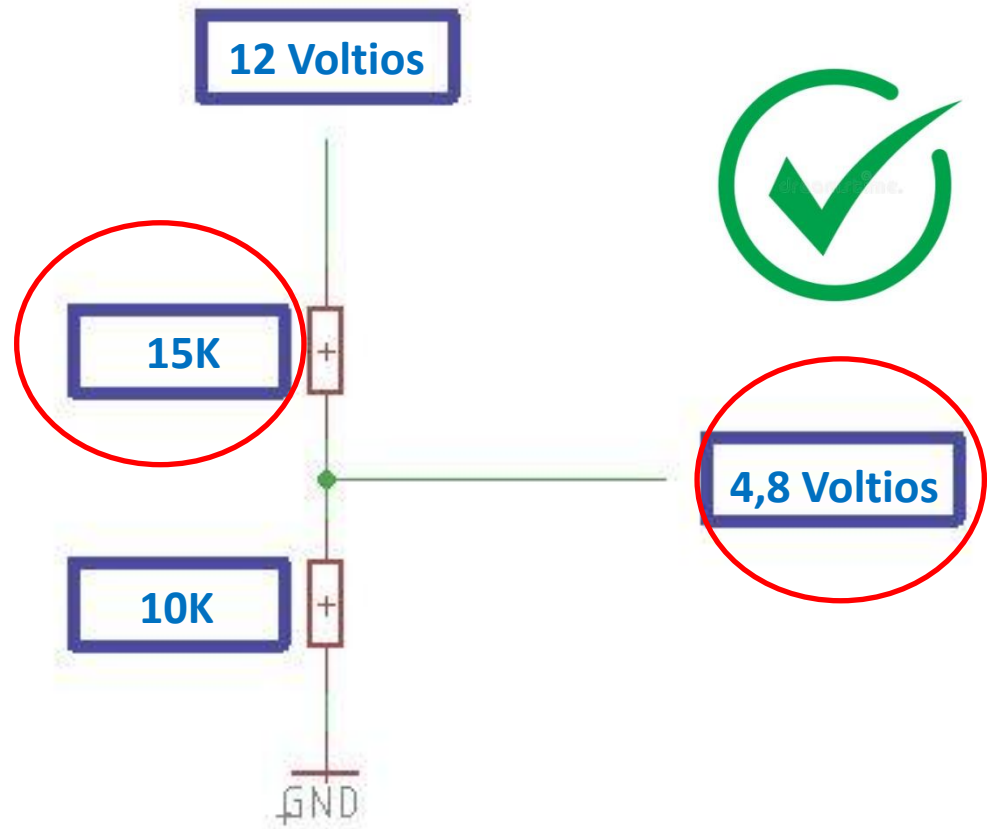
La resistencia comercial mas cercana que venden es la de 15K, por lo que veremos que pasa si la cambio por la que nos había salido en el calculo de 14K.

<i>Divisor resistivo</i>			
Tensión en Voltios	Resistencias Kilo ohmios	Corriente mili Amperios	Tensión en cada Resistencia
12	15		7,200
	10		4,800
	R total Kohm		
	25	0,480	12,000

Divisores de tensión

Como decíamos, la resistencia que hemos comprado es una de 15K, en lugar de la que nos daban nuestros cálculos de 14K, según el calculo de la página anterior la cosa quedaría así:

Como veis al aumentar la resistencia baja la tensión en la salida, eso es lógico, después del cambio, nos sigue quedando una tensión, que aunque no son los cinco voltios, es mayor que la mínima que definía el umbral de incertidumbre del que habíamos hablado al principio, por lo que el sistema, cada vez que apliquemos los doce voltios, entenderá, sin ninguna duda, que en su entrada digital hay un UNO, y por tanto funcionará como esperamos.



Entradas analógicas

No, si esto está muy bien, muy bonito, pero yo lo que quiero es medir mi batería con una entrada analógica, ¿y si no para que tanto rollo al principio con eso de que si la tensión es analógica o digital?, si señor, pero para llegar aquí necesitamos todo lo que hemos visto hasta ahora, ¿que ocurre si la entrada famosa de 12 voltios no es una señal que conecta o desconecta, sino que te da una tensión en función de la temperatura?, supongamos que se trata de una señal que te da cero voltios a cero grados y 12 voltios a cien grados, una vez que tenemos claro como funciona nuestra señal analógica de temperatura nos metemos en un poco de teoría.

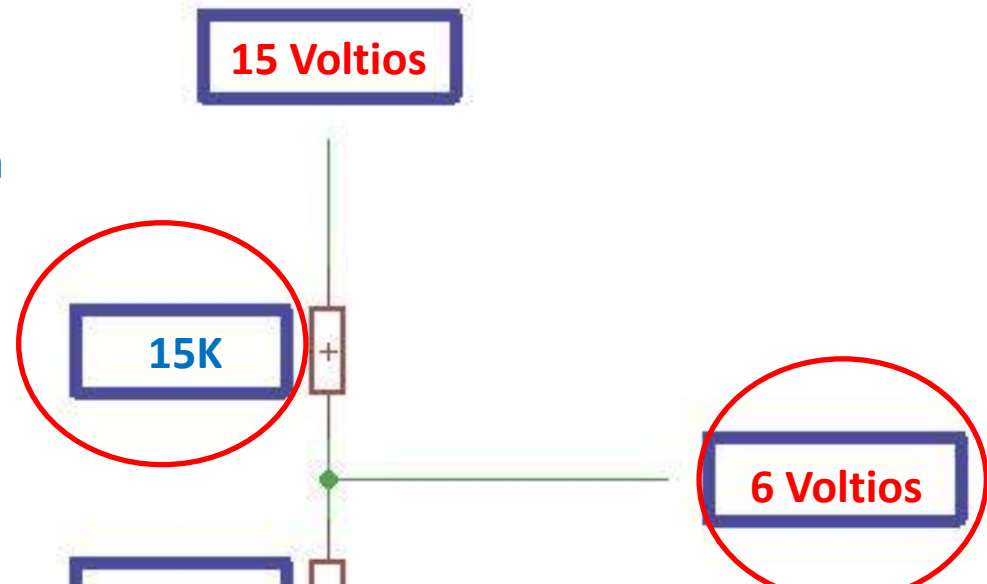
Las entradas analógicas, según su resolución y su referencia, convierten un valor analógico en un equivalente digital, la resolución es el numero bits, el numero de puntos en que va a dividir la señal analógica aplicada a la entrada, en el caso de Arduino, la resolución es de 10 bits, 2^{10} , o 1024 puntos, todo es lo mismo, esto significa, que una señal de cinco voltios se divide en 1024 trozos, por lo que cada trozo representa 0,0048828125 voltios, es decir, aproximadamente 5 milésimas de voltio por cada punto, por tanto si tenéis una señal que varía menos de 5 milésimas de voltio, el sistema no va a ser capaz de ver esa variación, esto funciona cuando el Arduino, tiene como referencia “DEFAULT”, esta es la que va a tener por defecto, si se define como “INTERNAL”, la referencia de 1024 puntos corresponderá a 1'1 Voltios, con esto cada punto es aproximadamente una milésima de voltio, se puede elegir incluso una referencia exterior, los comandos para estas dos referencias que he descrito son:

```
analogReference (INTERNAL) ; Y analogReference (DEFAULT) ;
```

Entradas analógicas

Bueno, pues vamos a ver que hacemos con esto para medir nuestra batería, ya que las entradas analógicas de Arduino consiguen unas medidas de considerable precisión y repetitibilidad, para empezar si vamos a medir una batería de 12 voltios, hemos de pensar que puede llegar hasta 13,8 voltios, por lo que si solo medimos hasta 12 perderemos información cuando mas la necesitamos

Ya que tenemos nuestro divisor, vamos a ver que pasa si subimos a 15 voltios, lo primero que vemos es que nos llegarían 6 voltios a una entrada que solo soporta 5 con lo que destruiríamos el Arduino o al menos esa entrada.



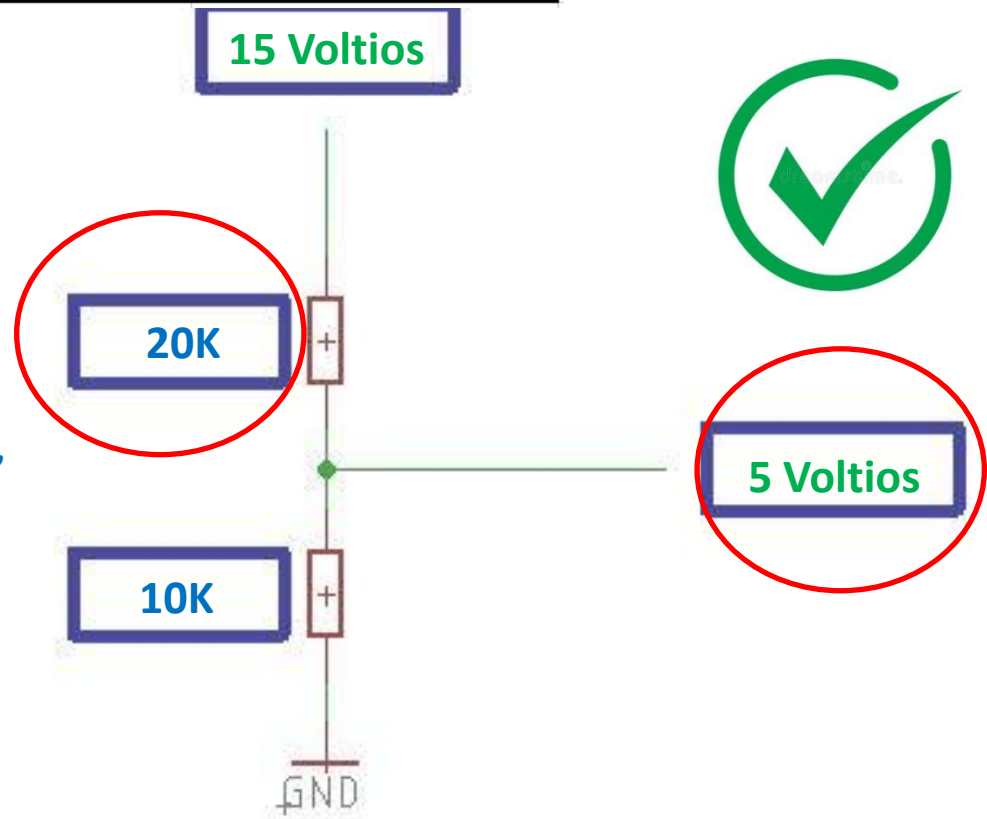
Divisor resistivo			
Tensión en Voltios	Resistencias Kilo ohmios	Corriente mili Amperios	Tensión en cada Resistencia
15	15		9,000
	10		6,000
	R total Kohm		
	25	0,600	15,000

Entradas analógicas

Si para 5 voltios poníamos 10K, para 10 voltios, hasta llegar a los 15 que queremos medir pondríamos 20K, parece lógico no?

Divisor resistivo			
Tensión en Voltios	Resistencias Kilo ohmios	Corriente mili Amperios	Tensión en cada Resistencia
15	20		10,000
	10		5,000
	R total Kohm		
	30	0,500	15,000

Nuestro calculo nos confirma que con estos valores de resistencia, cuando tengamos 15 voltios de entrada tendríamos 5 voltios en el Arduino, por lo que 15 voltios antes del divisor corresponderían al numero 1023, ¿como que 1023?, habías dicho 1024, si, el cero también cuenta, 1023 puntos y el cero, que también es un valor



Entradas analógicas

Una vez hecho este cambio y sabiendo que 15 voltios en la entrada corresponden a cinco voltios en el Arduino y que al mismo tiempo corresponde al valor analógico 1023, el programa que podíamos hacer para leer la tensión de batería es el siguiente:

```
TensionBateria = ((analogRead(Bateria) * 15.00) / 1023.00); //
```

TensionBateria es la variable donde vamos a tener la tensión de la batería medida, es una variable del tipo Float.

`analogRead(Bateria)` aquí Bateria es la entrada analógica, por ejemplo A1.

15 es la tensión para la que hemos diseñado el divisor.

1023 es la resolución.

De esta manera tan simple medimos la batería, pero como nunca nada es perfecto en esta vida de nuevo el fabricante nos fastidia la vida porque no hace resistencias de 20K, la mas cercana que hace es de 22 K, así que ponemos una resistencia de 22K con lo que mide de menos, menuda faena, ¿como lo arreglo?.

Entradas analógicas

Nos preguntábamos como arreglar el problema de que no tengan la resistencia exacta y yo tenga que poner otra, además las resistencias tienen una cosa que se llama tolerancia, esto significa que una resistencia de 10K, en realidad no mide 10K sino algo muy similar, así que a pesar de nuestros cálculos, y aunque fueran perfectos, siempre mediríamos de una manera aproximada por culpa de la tolerancia de las resistencias, pero no vamos a rendirnos, ¿os acordáis de nuestra línea de programa?:

```
TensionBateria = ((analogRead(Bateria) * 15.00) / 1023.00); //
```

Pues nos vamos a centrar en el numero 15.00, si al poner una resistencia mayor eso significa que estamos midiendo de menos, eso significa que para medir lo mismo deberíamos aplicar mayor tensión, ¿es así, verdad?, pues bien vamos a darle a esto el ajuste definitivo, suponed que medimos con un polímetro y hay 11 voltios, mientras el Arduino nos dice que hay 9'5, pues hacemos otra regla de tres directa, hacemos así, si a 9'5 voltios le corresponde 15, a 11 le corresponde x, o lo que es lo mismo: $(11 * 15) / 9'5 = 17'368$, cambiamos el numero 15 por este nuevo factor y la medida será perfecta, como muestra os pongo una línea del programa que controla mi barco:

```
TensionBateria = ((analogRead(Bateria) * 16.9651) / 1023.00); //
```

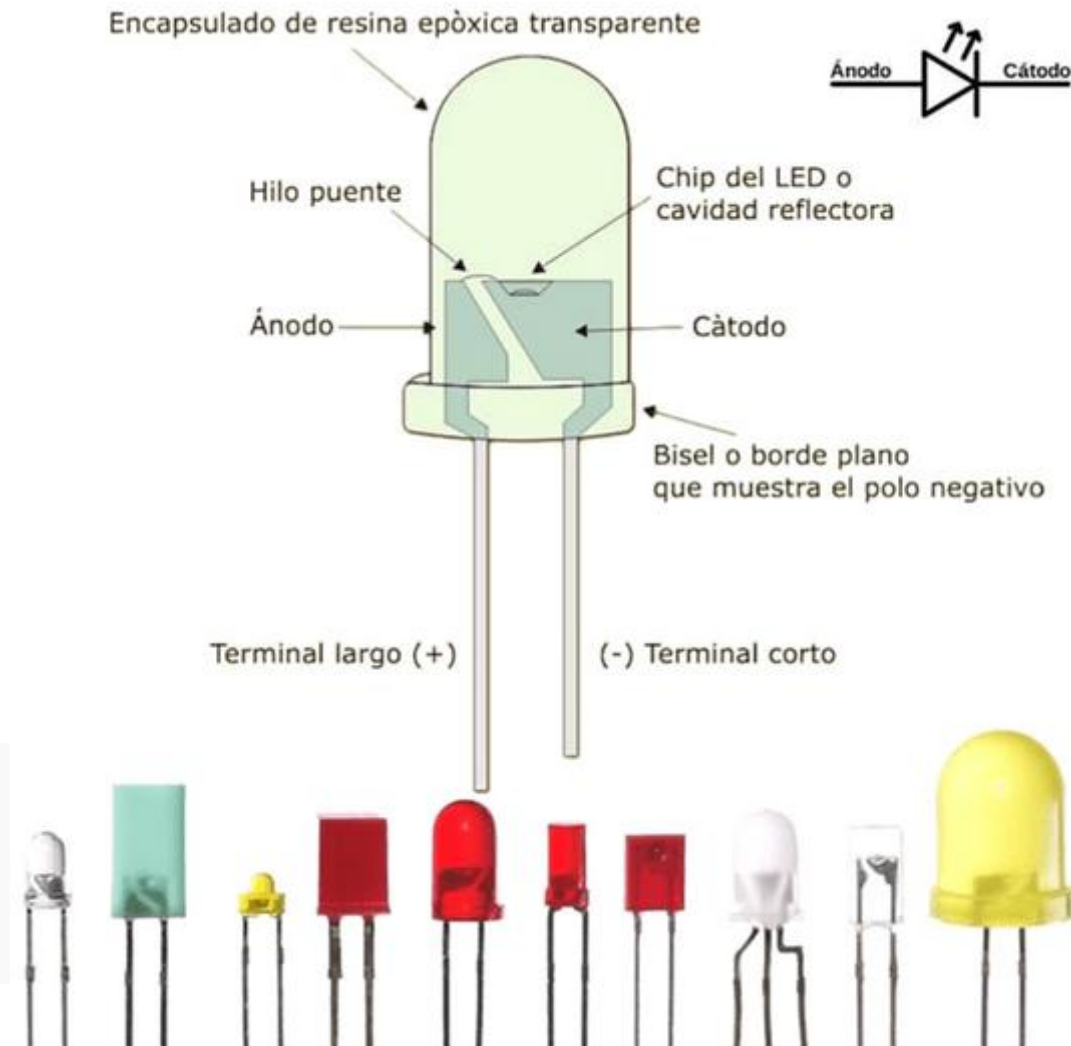
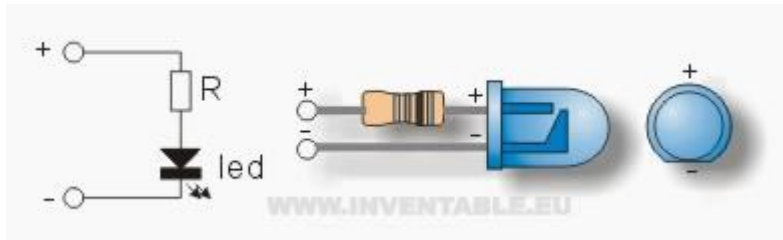
Como veis, el valor que yo tengo, y que tuve que calcular en su día, se asemeja bastante al que hemos calculado.

Salidas digitales

Si habéis llegado vivos hasta aquí, solo me cabe felicitaros, vamos a dedicarnos ahora a algo mucho mas simple, voy a encender un led, antiguamente la mayoría de los led eran de 1,6 voltios y de 16 miliamperios, a día de hoy, la mayoría son de una tensión cercana a los tres voltios, y una corriente de 20 miliamperios, vamos a hacer los cálculos para estos últimos, aunque perfectamente podéis usar los cálculos que os propongo para cualquier led o cualquier otro tipo de componente, si tenemos un led de 3 voltios y una tensión de 5 voltios, como la de las salidas del Arduino, o los 12 voltios de una batería, la manera de proceder siempre es la misma, tened en cuenta que lo importante en los led es la corriente que circula por ellos, ya que sobre todo en los modelos de mas potencia, la tensión del led varia con la temperatura que genera el mismo al funcionar, pero vamos a centrarnos en los led que normalmente usamos, el calculo de la resistencia para un led vale lo mismo para una salida digital, que para una salida analógica, siempre que se diseñe para la máxima tensión que la salida pueda proporcionar.

Encendiendo un led

Como todos sabéis un led tiene dos patas o terminales, el mas largo es el ánodo y mira hacia el positivo, el mas corto es el cátodo y mira hacia el negativo, los leds hay que montarlos con una resistencia de protección, que disipa la tensión que el led no necesita, del calculo de esa resistencia nos vamos a ocupar a continuación.



Encendiendo un led

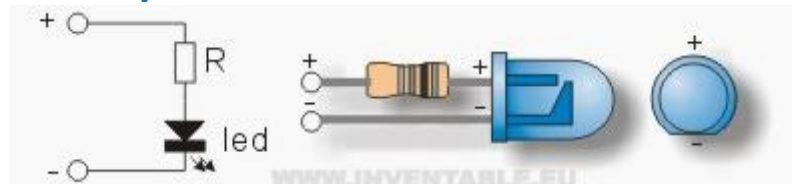
Si como habíamos dicho antes el led que tenemos es de tres voltios y tenemos una tensión de cinco voltios, y además queremos que circulen 20 miliamperios.

Lo primero que hacemos es restarle la tensión del led a la tensión total, es decir, 5 voltios, menos los 3 voltios del led son 2 voltios.

Si tenemos 2 voltios y queremos que circulen 20 miliamperios, o lo que es lo mismo, 0'02 Amperios, tenemos que usar la ley de Ohm, que nos dice que la corriente es proporcional a la tensión e inversamente proporcional a la corriente, es decir, que dividimos la tensión por la corriente y nos da la resistencia, así, en un momento:

$$2 \text{ voltios} / 0'02 \text{ Amperios} = 100 \text{ Ohmios}$$

Por tanto nuestra resistencia para un led de tres voltios funcionando a cinco voltios será de 100 Ohmios.



Lo mismo para un led de 3'2 voltios funcionando a 12 voltios:

$$12 - 3'2 = 8'8 \text{ Voltios} ; 8'8 / 0'02 = 440 \text{ Ohmios, esto es fácil,}$$

(aunque la resistencia de 440 Ohmios no se fabrique sino que la mas cercana es de 470 Ohmios)

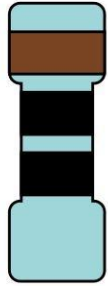
Valores de resistencia

Como me vais a decir que os amargo la vida con las resistencias que se fabrican y las que no, os dejo las tablas de los valores de resistencias que son comerciales

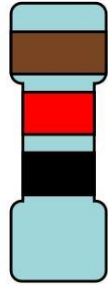
ROW	GOLD	BLACK	BROWN	RED	ORANGE	YELLOW	GREEN
1-	1R0	10R	100R	1K0	10K	100K	1M0
2-	1R1	11R	110R	1K1	11K	110K	1M1
3-	1R2	12R	120R	1K2	12K	120K	1M2
4-	1R3	13R	130R	1K3	13K	130K	1M3
5-	1R5	15R	150R	1K5	15K	150K	1M5
6-	1R6	16R	160R	1K6	16K	160K	1M6
7-	1R8	18R	180R	1K8	18K	180K	1M8
8-	2R0	20R	200R	2K0	20K	200K	2M0
9-	2R2	22R	220R	2K2	22K	220K	2M2
10-	2R4	24R	240R	2K4	24K	240K	2M4
11-	2R7	27R	270R	2K7	27K	270K	2M7
12-	3R0	30R	300R	3K0	30K	300K	3M0
13-	3R3	33R	330R	3K3	33K	330K	3M3
14-	3R6	36R	360R	3K6	36K	360K	3M6
15-	3R9	39R	390R	3K9	39K	390K	3M9
16-	4R3	43R	430R	4K3	43K	430K	4M3
17-	4R7	47R	470R	4K7	47K	470K	4M7
18-	5R1	51R	510R	5K1	51K	510K	5M1
19-	5R6	56R	560R	5K6	56K	560K	5M6
20-	6R2	62R	620R	6K2	62K	620K	6M2
21-	6R8	68R	680R	6K8	68K	680K	6M8
22-	7R5	75R	750R	7K5	75K	750K	7M5
23-	8R2	82R	820R	8K2	82K	820K	8M2

La letra R entre dos números se usa como si fuese una coma, la letra K o la letra M que significa Mega hace lo mismo

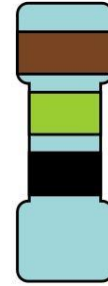
Valores de resistencia



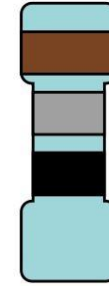
10



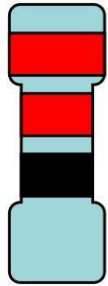
12



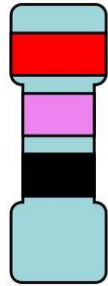
15



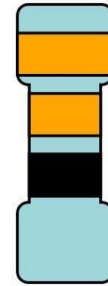
18



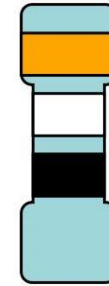
22



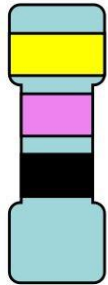
27



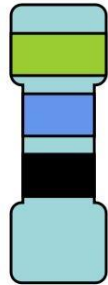
33



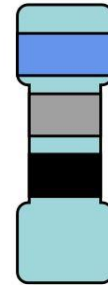
39



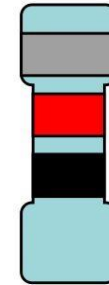
47



56



68

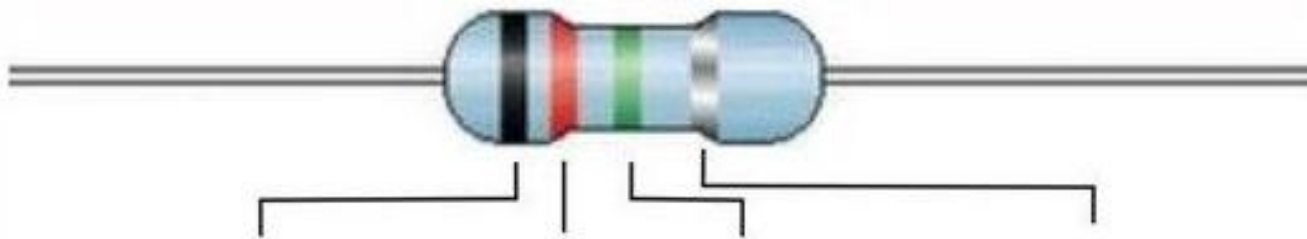


82

Estos son los dos primeros números de la resistencia, estos valores se multiplican o dividen por 10, 100, 1000, etc.

Valores de resistencia

Esta es la manera de leer el valor de una resistencia de 4 bandas, la tercera banda es el numero de ceros.



Color	1ra. Banda	2da. Banda	3ra. Banda Multiplicador	Tolerancia %
Negro	0	0	x1	
Cafe	1	1	x10	
Rojo	2	2	x100	2%
Naranja	3	3	x1000	
Amarillo	4	4	x10000	
Verde	5	5	x100000	
Azul	6	6	x1000000	
Violeta	7	7	x10000000	
Gris	8	8	x100000000	
Blanco	9	9	x1000000000	
				Dorado 5%
				Plata 10%

Aquí también veis como aparece la tolerancia de la que hablábamos antes

Valores de resistencia

También hay resistencias con mayor numero de bandas que nos dan mas información

<p>0 1 2 3 4 5 6 7 8 9</p> <p>0 Negro</p> <p>1 Marrón</p> <p>2 Rojo</p> <p>3 Naranja</p> <p>4 Amarillo</p> <p>5 Verde</p> <p>6 Azul</p> <p>7 Purpura</p> <p>8 Gris</p> <p>9 Blanco</p> <p>±1% Marrón</p> <p>±2% Rojo</p> <p>±5% Dorado</p> <p>±10% Plateado</p>	<p>±1%</p> <p>±2%</p> <p>±5%</p> <p>±10%</p> <p>1.5K</p> <p>0 X1</p> <p>1 1 X10</p> <p>2 2 X100</p> <p>3 3 X1000</p> <p>4 4 X10000</p> <p>5 5 X100000</p> <p>6 6 X1000000</p> <p>7 7 ÷10</p> <p>8 8 ÷100</p> <p>9 9</p>	<p>±1%</p> <p>±2%</p> <p>±5%</p> <p>±10%</p> <p>15K</p> <p>0 0 X1</p> <p>1 1 1 X10</p> <p>2 2 2 X100</p> <p>3 3 3 X1000</p> <p>4 4 4 X10000</p> <p>5 5 5 ÷10</p> <p>6 6 6 ÷100</p> <p>7 7 7</p> <p>8 8 8</p> <p>9 9 9</p>	<p>±1%</p> <p>±2%</p> <p>±5%</p> <p>±10%</p> <p>100 50</p> <p>25 15</p> <p>10 5</p> <p>1 PPM</p> <p>620K</p> <p>0 0 X1</p> <p>1 1 1 X10</p> <p>2 2 2 X100</p> <p>3 3 3 X1000</p> <p>4 4 4 X10000</p> <p>5 5 5 ÷10</p> <p>6 6 6 ÷100</p> <p>7 7 7</p> <p>8 8 8</p> <p>9 9 9</p>
Código de Colores	Resistencias de 4 Bandas	Resistencias de 5 Bandas	Resistencias de 6 Bandas

Veréis que cada color siempre representa al mismo numero

Ley de Ohm

La base de toda la electrónica y la electricidad es la ley de Ohm, con unas simples formulas es sencillo calcular la mayoría de los parámetros de un circuito eléctrico o electrónico, aquí os dejo las formulas:

Tensión	=	Corriente	x	Resistencia
Corriente	=	Tensión	/	Resistencia
Resistencia	=	Tensión	/	Corriente

Potencia	=	Tensión	x	Corriente
Tensión	=	Potencia	/	Corriente
Corriente	=	Potencia	/	Tensión

