

Introducción a la Plataforma Freedom

Ing. Gerardo Carmona

Recursos

Principales Recursos

- Hoja de especificaciones :: Fabricante Freescale, modelo KL25Z
- mbed.org :: Página oficial del compilador
- makerobots.tk :: Página web personal con contenido educativo

Clase 1

1. ¿Qué es un microcontrolador?
2. Hardware
3. Programación de la tarjeta
4. Software
5. Primer programa
6. Led RGB

KL25Z

Contenido del día

¿Qué es un
microcontrolador?

Microcontrolador

Es un circuito integrado que contiene muchas de las mismas cualidades que una computadora de escritorio, tales como la CPU (Unidad Central de Procesamiento), la memoria, etc., pero no incluye ningún dispositivo de “comunicación con humanos”, como monitor, teclados o mouse.

Los microcontroladores son diseñados para aplicación de control de máquinas, más que para interactuar con humanos.

¿Dónde los encuentro?



Hardware

Introducción a la Freedom KL25Z

Es una tarjeta de desarrollo la cual contiene “algunos” componentes electrónicos para poder empezar a programarla sin conectar nada adicional.

Cuenta además con diversos dispositivos como lo son:

- Led RGB cátodo común
- Acelerómetro de 3 ejes
- Sensor slider capacitivo
- USB host

Pines de E/S

Contiene diversos pines E/S digitales

(24) Salidas PWM

(5) Entradas analógicas

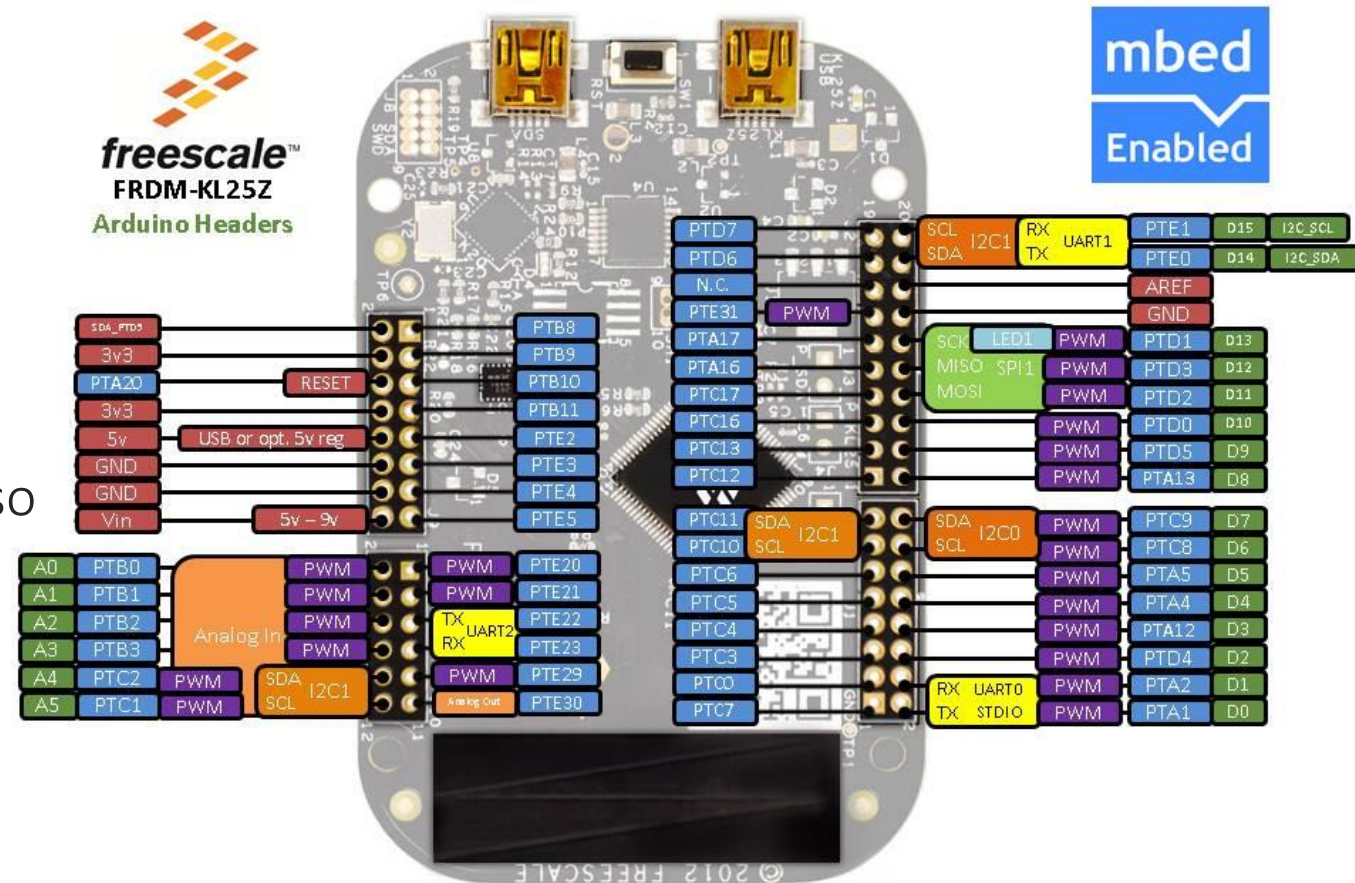
(2) Puertos de comunicación Serie (tx, rx)

(2) Puertos de comunicación I2C (scl, sda)

(1) Puertos de comunicación SPI (mosi, miso y sck)

(1) Salida analógica

(1) Puerto de comunicación USB-Serie



Software

Programación de la tarjeta

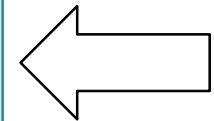
Existen diversos métodos para poder programar la tarjeta Freedom, entre los cuales destacan 2:

Code Warrior	mbed.org
Programador nativo	Programador en web
Configuras a detalle cada una de las características de la tarjeta	Contiene librerías de código que facilitan su programación (características por default)
Tiempo de desarrollo largo	Tiempo de desarrollo corto
Documentación moderada, pero difícil de encontrar	Documentación moderada
Licencia educacional limitada a cierto número de líneas por programa	Si no hay internet, no puedes trabajar, pero gratis

Programación de la tarjeta

Existen diversos métodos para poder programar la tarjeta Freedom, entre los cuales destacan 2:

Code Warrior	mbed.org
Programador nativo	Programador en web
Configuras a detalle cada una de las características de la tarjeta	Contiene librerías de código que facilitan su programación (características por default)
Tiempo de desarrollo largo	Tiempo de desarrollo corto
Documentación moderada, pero difícil de encontrar	Documentación moderada
Licencia educacional limitada a cierto número de líneas por programa	Si no hay internet, no puedes trabajar, pero gratis



Cambiando el bootloader

Primer paso es cambiar el bootloader de la tarjeta

La tarjeta de fabrica tiene un código que facilita la programación mediante el Code Warrior y hay que cambiar esto para poderla programar usando el compilador web.

Sigue los siguientes pasos

1. Conecta el cable usb en tu computadora
2. Presiona el botón de “Reset” de la tarjeta y no lo sueltes
3. Conecta el cable usb mientras mantienes presionado el botón de “Reset” del lado que dice “SDA”
4. Suelta el botón de “Reset”
5. Se instalará un nuevo dispositivo
6. Aparecerá como un dispositivo de almacenamiento masivo en “Equipo” con el nombre de “Bootloader”



Cambiar el bootloader

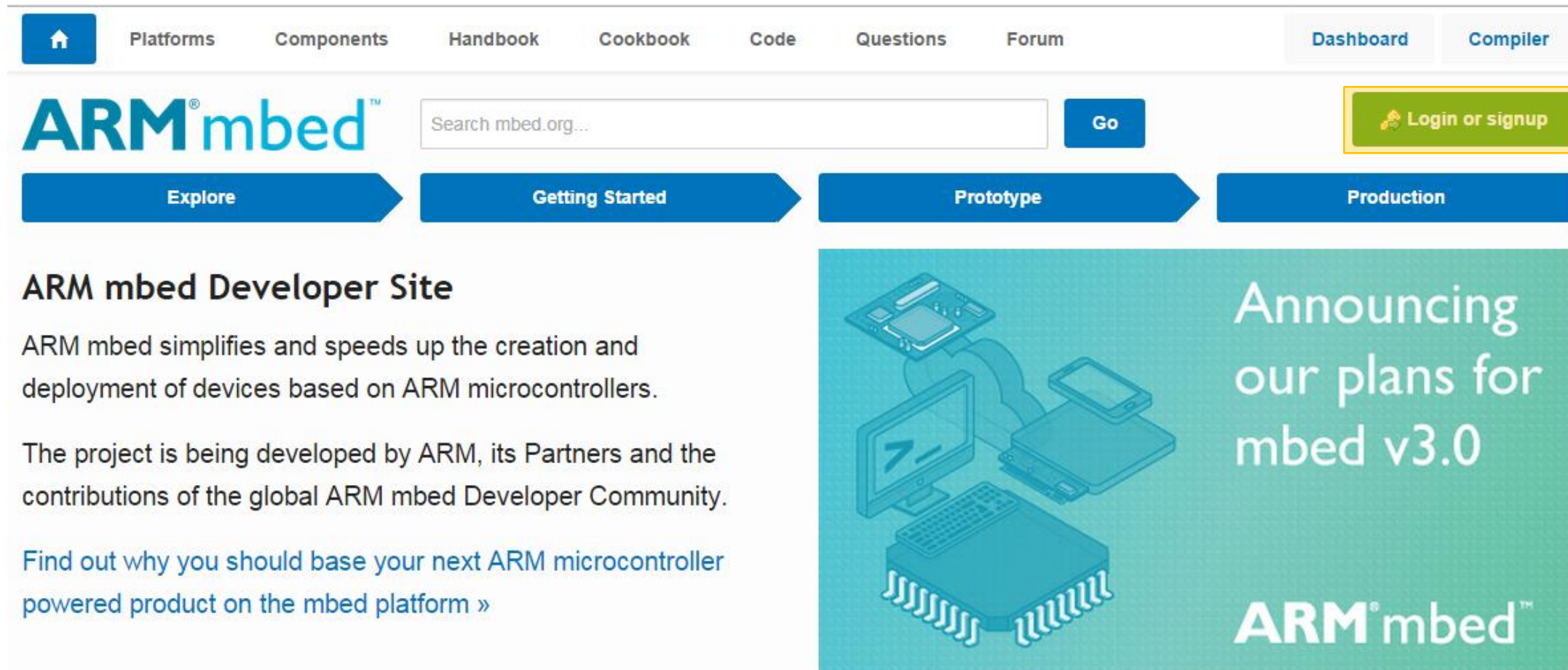
1. Entra a la página “mbed.org”
2. Da clic en el menú “Plataforms”
3. Selecciona la “Freescale KL25Z”
4. Busca el link “mbed_if_v2.0_frdm_kl25Z”
5. Clic derecho sobre el link
6. Clic “Guardar enlace como...” si estas usando Chrome
7. Guarda el archivo dentro de tu tarjeta
8. Desconecta y conecta tu cable usb, deberás de aparecer un nuevo dispositivo llamado mbed

Práctica 1

Parpadeo de un Led

Crear un programa nuevo

Registra una cuenta en developer.mbed.org, es gratis.

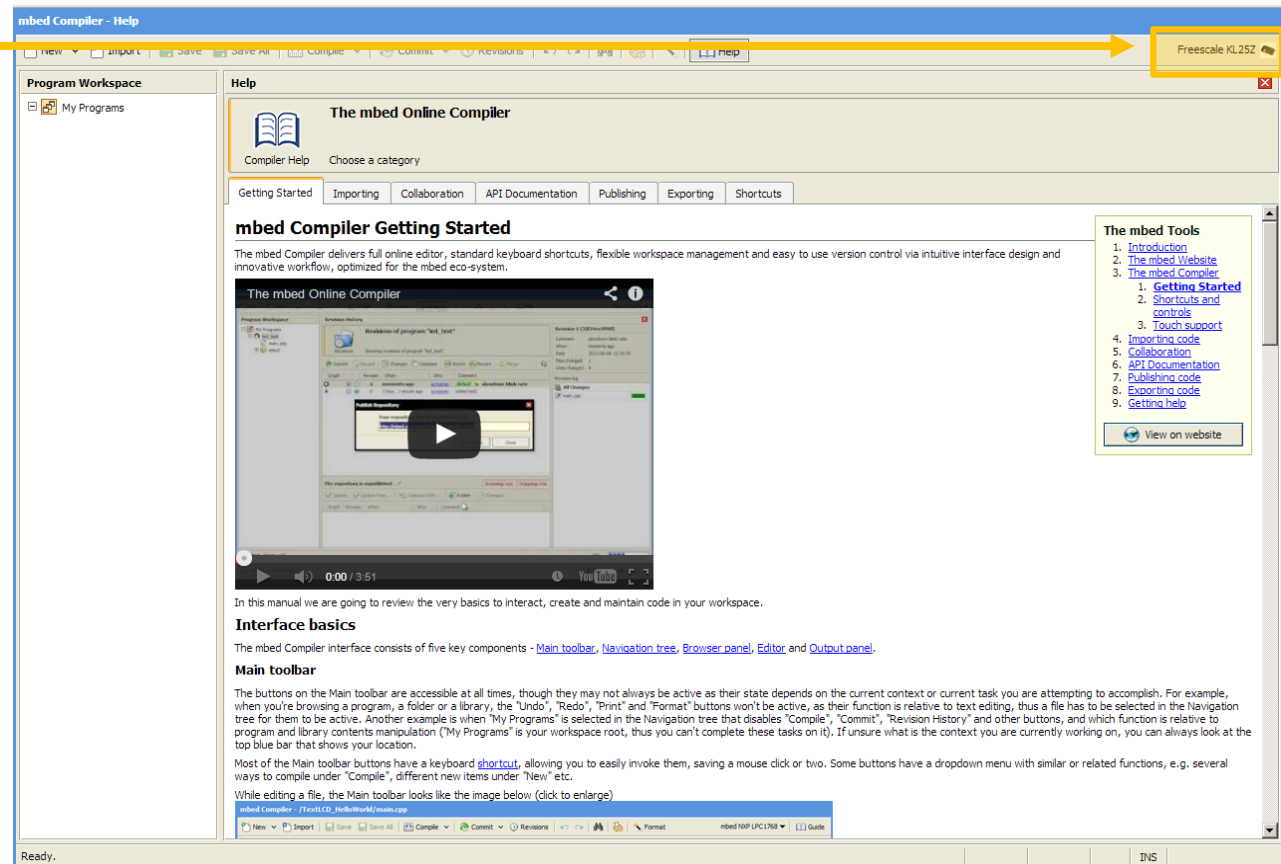
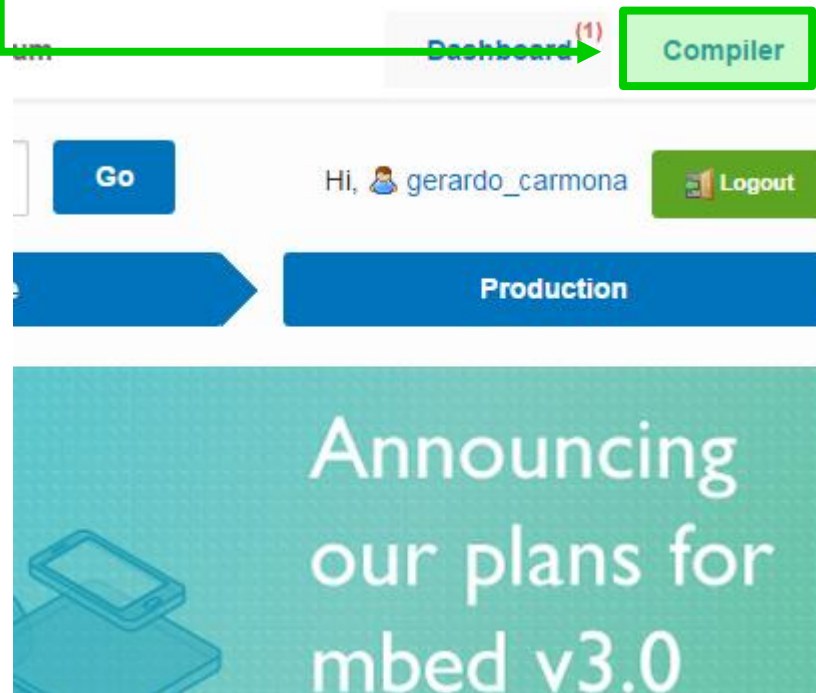


The screenshot shows the ARM mbed Developer Site homepage. At the top, there is a navigation bar with links for Home, Platforms, Components, Handbook, Cookbook, Code, Questions, and Forum. On the right side of the navigation bar, there are buttons for Dashboard and Compiler. Below the navigation bar, the ARM mbed logo is displayed on the left, and a search bar with the placeholder text "Search mbed.org..." and a "Go" button is in the center. To the right of the search bar is a green button with a user icon and the text "Login or signup". Below the search bar, there are four blue arrow-shaped buttons labeled "Explore", "Getting Started", "Prototype", and "Production". The main content area features a section titled "ARM mbed Developer Site" with a sub-header "ARM mbed simplifies and speeds up the creation and deployment of devices based on ARM microcontrollers." Below this, there is a paragraph stating "The project is being developed by ARM, its Partners and the contributions of the global ARM mbed Developer Community." and a link "Find out why you should base your next ARM microcontroller powered product on the mbed platform »". On the right side of the main content area, there is a large green banner with the text "Announcing our plans for mbed v3.0" and the ARM mbed logo at the bottom right. The banner also features an illustration of various electronic devices and a microcontroller chip.

Crear un programa nuevo

Clic en “compiler”









Clic en “no device selected”



Crear un programa nuevo


Selecciona la plataforma a utilizar.

Platforms

 <p>mbed LPC1768</p> <ul style="list-style-type: none">• Cortex-M3, 96MHz• 512KB Flash, 32KB RAM	 <p>mbed LPC1114U24</p> <ul style="list-style-type: none">• Cortex-M0, 48MHz• 32KB Flash, 8KB RAM	 <p>FRDM-KL25Z</p> <ul style="list-style-type: none">• Cortex-M0+• 128KB Flash, 16KB RAM• USB OTG	 <p>NXP LPC800-MAX</p> <ul style="list-style-type: none">• Cortex-M0+• 16KB Flash, 4KB RAM
			


Después clic en


Platform Partner



Freescale

Freescale is a leader in embedded processing solutions for the automotive, consumer, industrial and networking markets.

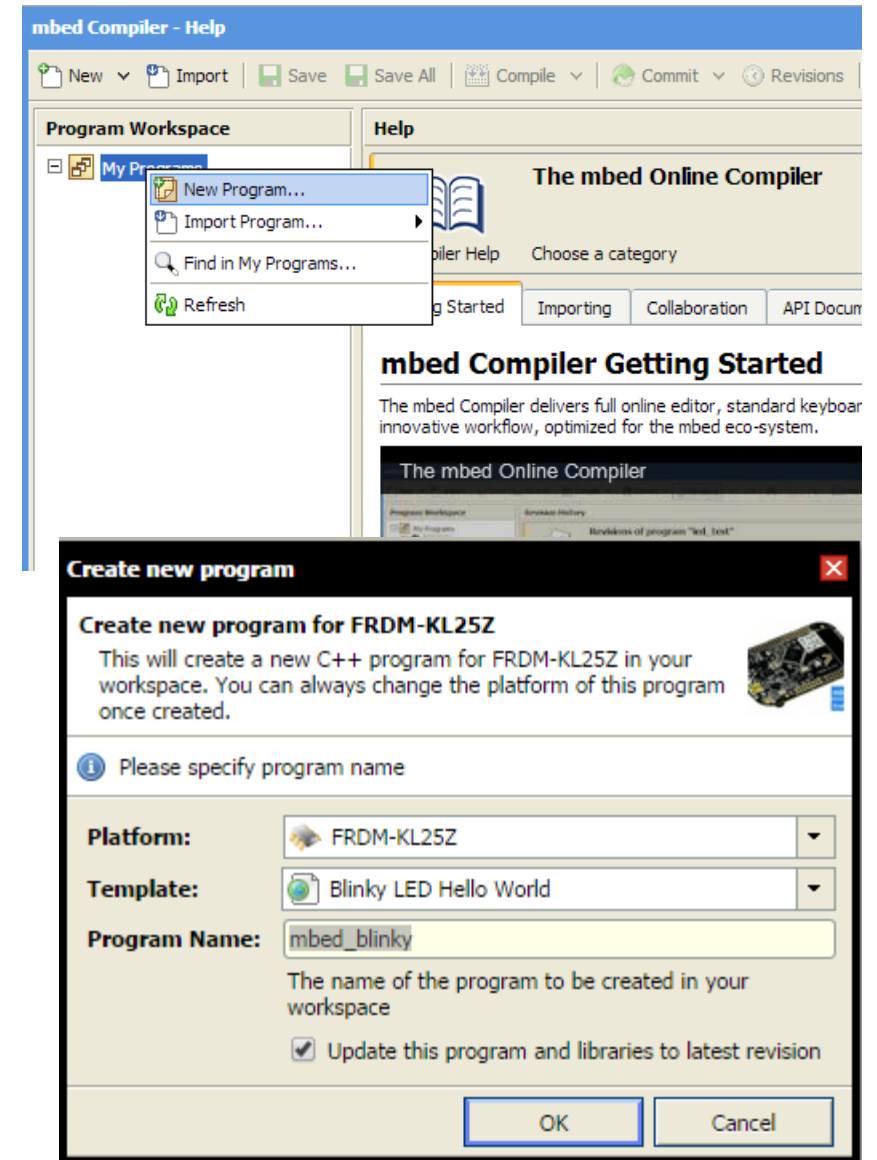
 **Add to your mbed Compiler**

 **Buy Now**

Volvemos a abrir el "Compiler"

Crear un programa

1. Clic derecho sobre “My programs”
2. Selecciona “New Program...”
3. Escribe el nombre de tu programa.
 - Nombres sin espacios ni puntos o caracteres extraños
 - Se recomiendan nombres descriptivos
 - Usualmente 2 palabras se separan con un “_”
4. En “Plataform” asegúrate que esté tu tarjeta
5. Verifica que en “Template” esté seleccionado “Blinky LED Hello World”.

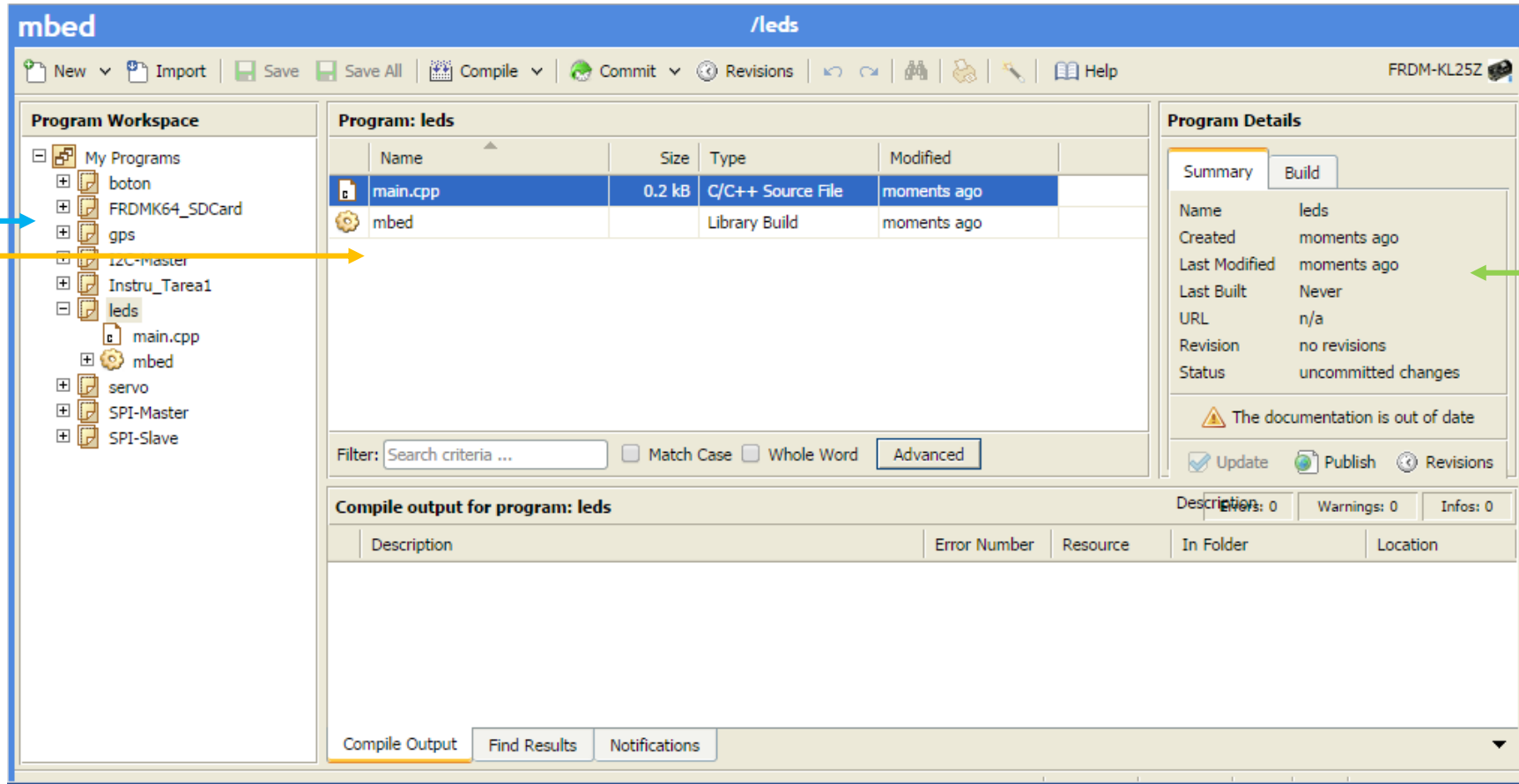


Crear un programa

Lista de tus programas

Contenido de tu programa seleccionado (leds)

Propiedades del elemento seleccionado



Leds

Librería mbed, todo programa debe contener esta línea de código. Esta manda llamar todas las funciones de mbed

Ciclo condicional (infinito)

```
main.cpp X
1 #include "mbed.h"
2
3 DigitalOut myled(LED1);
4
5 int main() {
6     while(1) {
7         myled = 1;
8         wait(0.2);
9         myled = 0;
10        wait(0.2);
11    }
12 }
13
```

Se declara un "objeto" llamado myled que pertenece a la clase DigitalOut y se envía el parámetro "LED1"

Programa principal, se ejecuta una sola vez, todo programa debe contenerlo.

Código que se ejecutará por una y otra vez.

No olvides cerrar todas las llaves. Estas indican donde empieza una función, un ciclo, un condicional, etc.

Leds

Mbed library, all programs must have this line of code in order to call the special functions from mbed.

Infinite cycle

```
main.cpp X
1 #include "mbed.h"
2
3 DigitalOut myled(LED1);
4
5 int main() {
6     while(1) {
7         myled = 1;
8         wait(0.2);
9         myled = 0;
10        wait(0.2);
11    }
12 }
13
```

It declares an "object" called "myled" that's part of the "class" "DigitalOut" and it sends the argument "LED1".

Main program, it executes only one time.

Code that is going to execute

No olvides cerrar todas las llaves. Estas indican donde empieza una función, un ciclo, un condicional, etc.

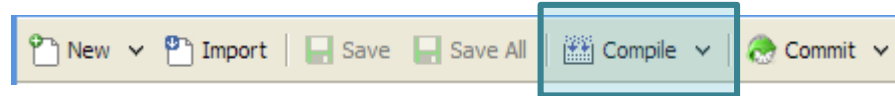
Nombres a los Pines

Mbed le puso nombres amistosos a ciertos pines, como al led RGB.

Color	Nombre 1	Nombre 2	Pin
Azul	LED3	LED_BLUE	PTD1
Verde	LED2	LED_GREEN	PTB19
Rojo	LED1	LED_RED	PTB18

Descargar y Programar

1. Clic en “Compile”



2. Si no hay errores de compilación te pedirá descargar un archivo, puedes descargarlo a tu computadora o directamente al dispositivo mbed (Freedom).
3. Si lo bajaste a la computadora copia y pega el archivo dentro del dispositivo mbed
4. Presiona el botón de “Reset” para iniciar tu programa

Led RGB

Ánodo común

Led RGB

Ánodo común



Led RGB

Led: Diodo emisor de Luz

RGB: Red-Green-Blue

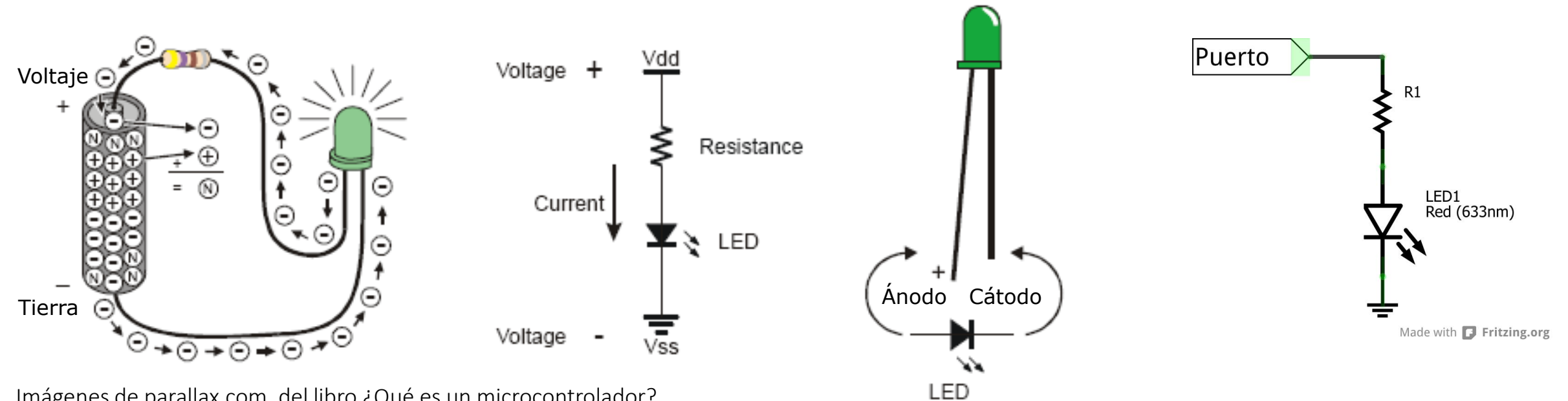
Ánodo Común: Que comparten la alimentación

Led

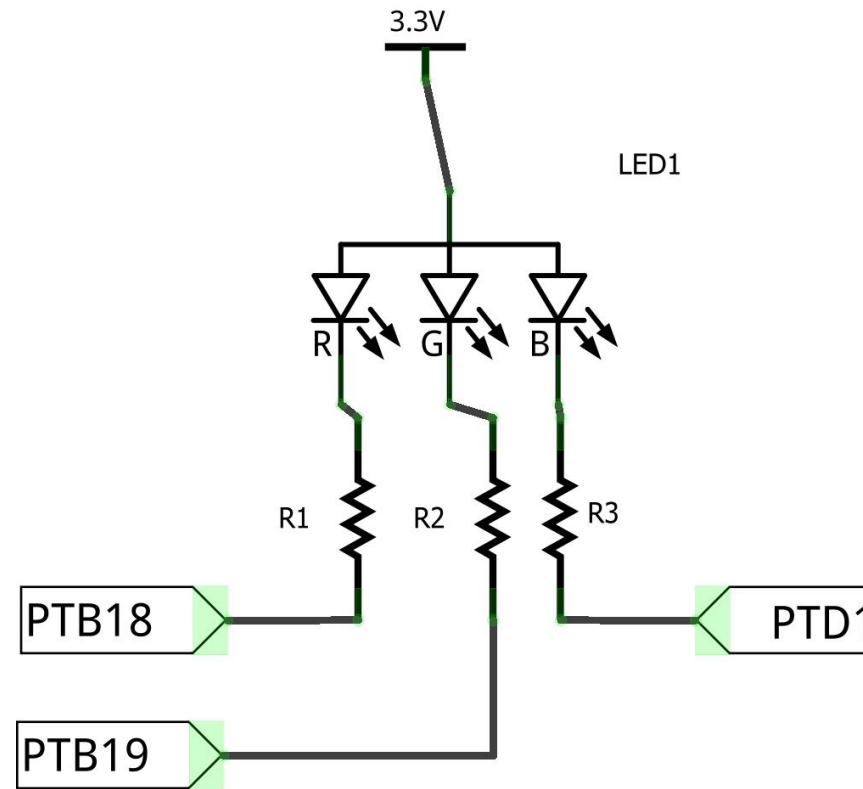
Se debe de conectar correctamente el led para que funcione.

Tres formas de determinar la terminal “+”

Un pin es capaz de mandar voltaje o tierra



Funcionamiento Led RGB



Nomenclatura

Nombre	Otros
Tierra	GND, 0V, lado negativo de la batería
Voltaje	V+, 3.3V, 5V, lado positivo de la batería
Ánodo	Lado positivo del led
Cátodo	Lado negativo del led

Práctica 2

Enciende cada led del RGB (Colores: Rojo, Verde, Azul)

Práctica 3

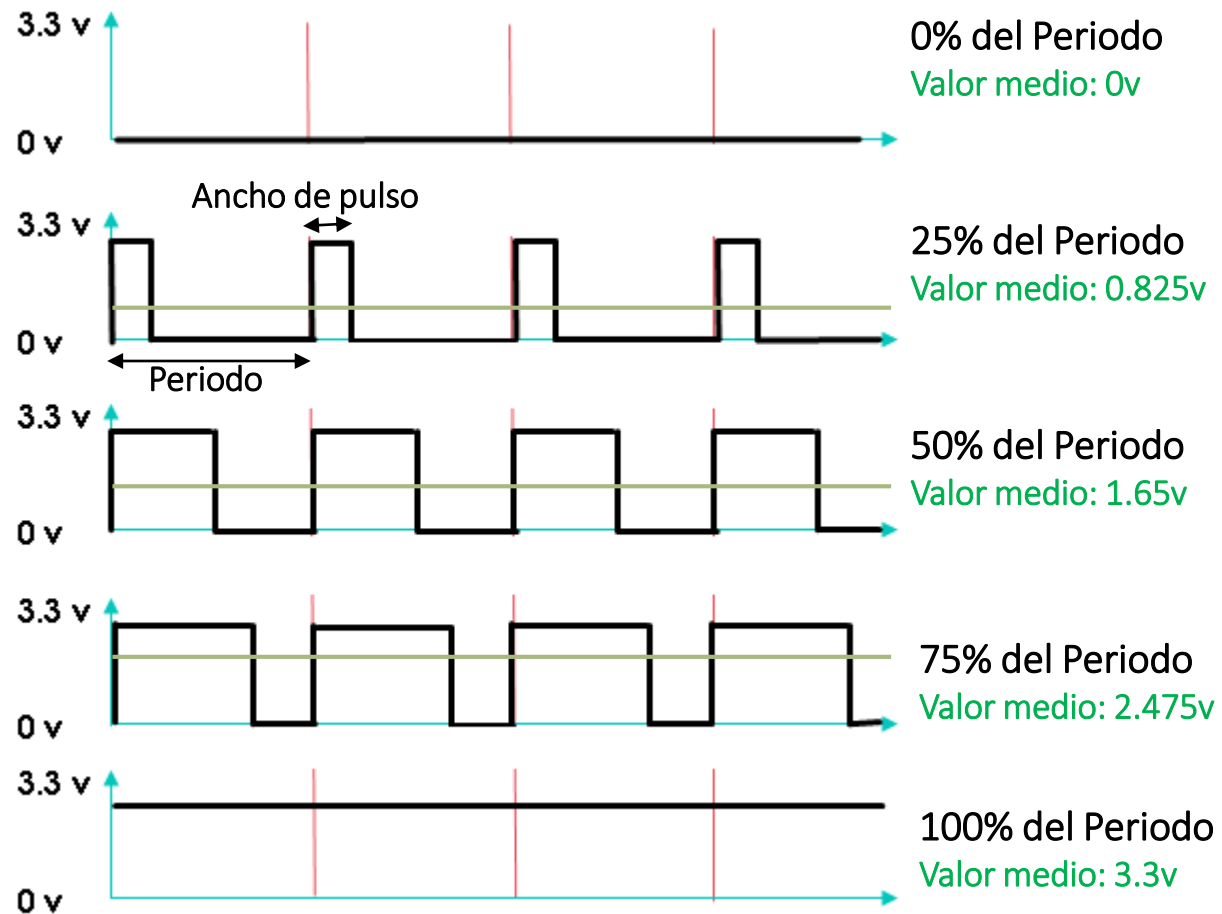
Prende las 7 combinaciones de colores posibles y modifica el tiempo de encendido a 500 ms

Señales PWM

Introducción a señales PWM

Señales PWM

PWM (Pulse Width Modulation)



Práctica 4

Ciclos “For”. Modifica tu programación para que los leds enciendan y apaguen modificando el brillo

Clase 2

1. Video Cubo de Leds
2. Agregar Hardware
3. Práctica 5: Semáforo con Leds
 1. Conexiones electrónicas
 2. Diseño electrónico
 3. Resistencias
4. Práctica 6: Cruce Peatonal
 1. Entradas digitales: botón normalmente abierto
 2. Resistencias internas

KL25Z

Contenido del día

Video



Agregar Hardware

Práctica 5

Semáforo

Semáforo con Leds

En esta práctica armaremos y programaremos un semáforo que siga la siguiente secuencia:

1. Verde por 3 segundos
2. Parpadea verde 3 veces (500 ms encendido, 500 ms apagado)
3. Prende amarillo por un segundo
4. Prende rojo por 3 segundos

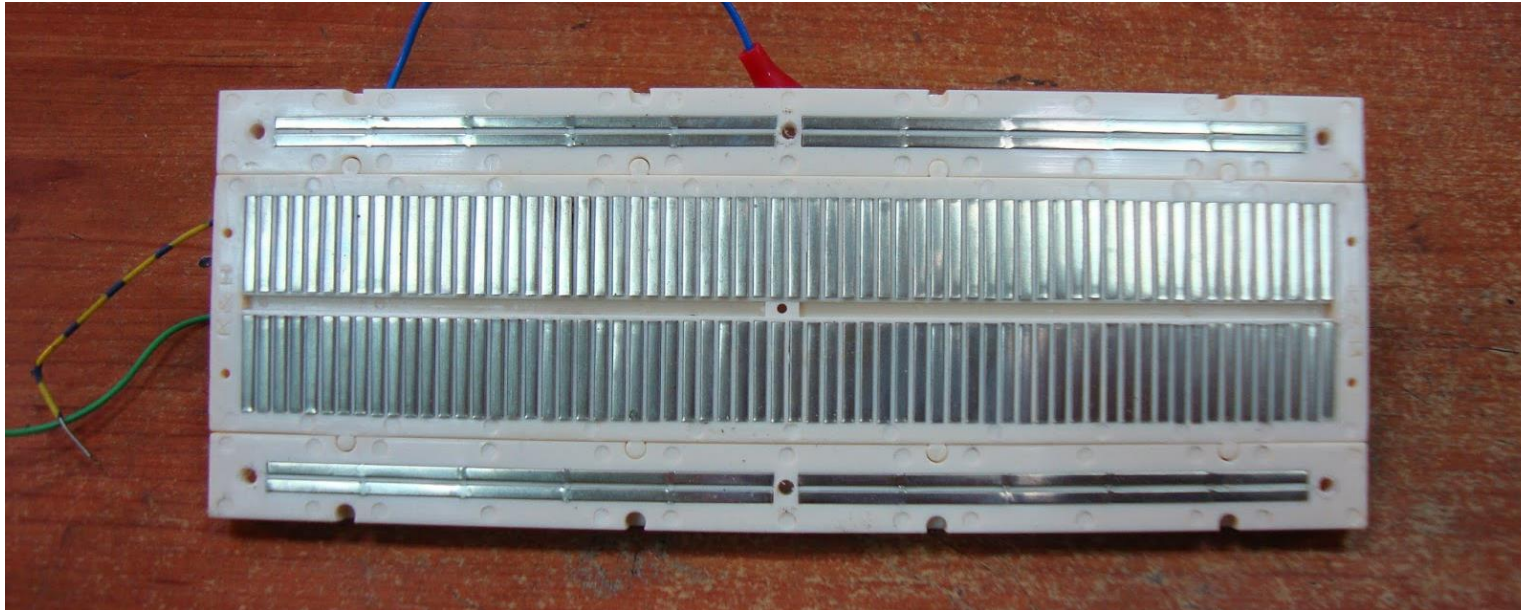
Recuerda que el semáforo no debe de estar apagado en ningún momento (excepto cuando el verde está parpadeando).

No hay 2 luces prendidas al mismo tiempo.



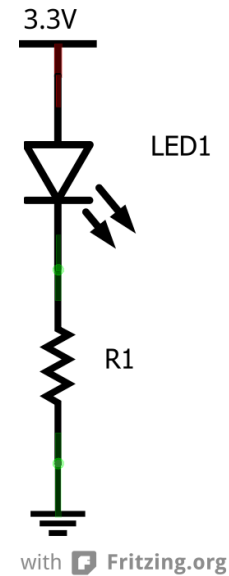
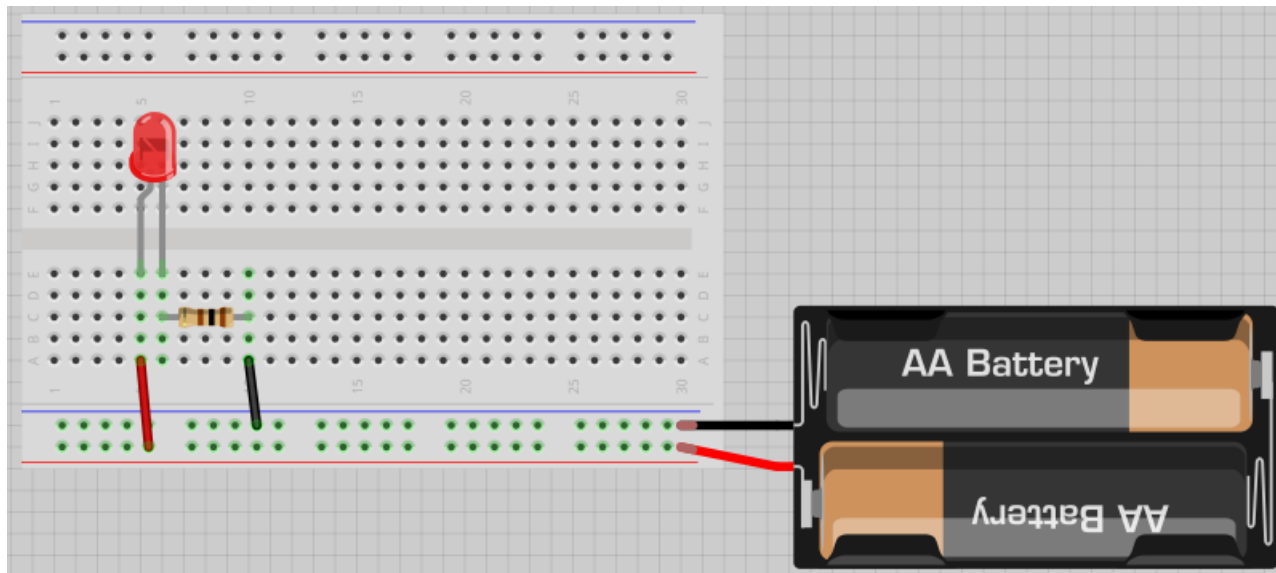
Protoboard

Sirve para hacer conexiones temporales de varios elementos electrónicos.



Conexiones Electrónicas

La imagen de la derecha muestra como conectar un led a una fuente de alimentación, el diagrama de la derecha es la representación electrónica de la imagen.



Diseño Electrónico

Necesitamos calcular el valor correcto de la resistencia a emplear.

Sabemos que $V = I * R$

V: Voltaje

I: Corriente eléctrica

R: Resistencia

La suma de todos los voltajes en un circuito cerrado es cero

$$V_{dd} - V_L - V_R = 0$$

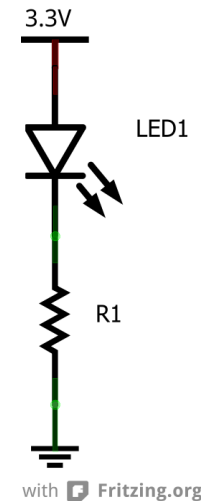
V_{dd} : Voltaje de alimentación

V_L : Voltaje led

V_R : Voltaje de la resistencia

Voltajes suministrados son positivos, voltajes consumidos son negativos

El voltaje de salida de los puertos es de 3.3V



Diseño Electrónico

Cálculos para la resistencia que necesitamos:

$$V_{CC} - V_L - V_R = 0$$

$V = I * R$

$$3.3 - 2 - 0.02 * R = 0$$
$$R = \frac{1.3}{0.02} = \boxed{65 \text{ ohms}}$$

LED color rojo Datos del fabricante	
Voltaje de consumo	2 Volts
Corriente máxima	20 mA

1 Ampere = 1,000 mA

Selección de la Resistencia

Tabla comercial de resistencias

x 1	x 10	x 100	x 1.000 (K)	x 10.000 (10K)	x 100.000 (100K)	x 1.000.000 (M)
1 Ω	10 Ω	100 Ω	1 KΩ	10 KΩ	100 KΩ	1 M Ω
1,2 Ω	12 Ω	120 Ω	1K2 Ω	12 KΩ	120 KΩ	1M2 Ω
1,5 Ω	15 Ω	150 Ω	1K5 Ω	15 KΩ	150 KΩ	1M5 Ω
1,8 Ω	18 Ω	180 Ω	1K8 Ω	18 KΩ	180 KΩ	1M8 Ω
2,2 Ω	22 Ω	220 Ω	2K2 Ω	22 KΩ	220 KΩ	2M2 Ω
2,7 Ω	27 Ω	270 Ω	2K7 Ω	27 KΩ	270 KΩ	2M7 Ω
3,3 Ω	33 Ω	330 Ω	3K3 Ω	33 KΩ	330 KΩ	3M3 Ω
3,9 Ω	39 Ω	390 Ω	3K9 Ω	39 KΩ	390 KΩ	3M9 Ω
4,7 Ω	47 Ω	470 Ω	4K7 Ω	47 KΩ	470 KΩ	4M7 Ω
5,1 Ω	51 Ω	510 Ω	5K1 Ω	51 KΩ	510 KΩ	5M1 Ω
5,6 Ω	56 Ω	560 Ω	5K6 Ω	56 KΩ	560 KΩ	5M6 Ω
6,8 Ω	68 Ω	680 Ω	6K8 Ω	68 KΩ	680 KΩ	6M8 Ω
8,2 Ω	82 Ω	820 Ω	8K2 Ω	82 KΩ	820 KΩ	8M2 Ω
						10M Ω

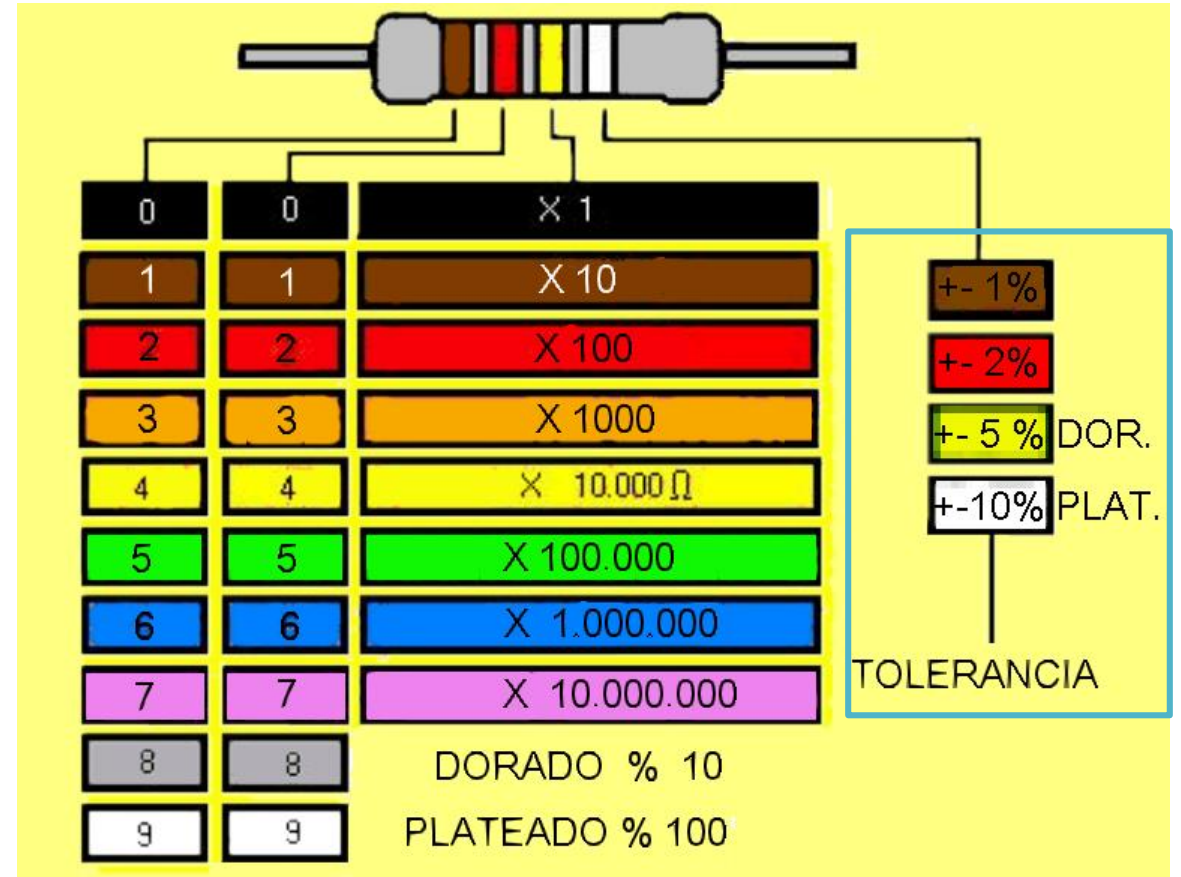
Resistencias

- Es un elemento que se resiste al flujo de la corriente eléctrica.
- Se mide en ohms (Ω)
- Código de colores

Resultado 65 ohms

Mas cercana 68 ohms
con error de +-5%

64.6 a 71.4 ohms



Selección de la Resistencia

Tabla comercial de resistencias

x 1	x 10	x 100	x 1.000 (K)	x 10.000 (10K)	x 100.000 (100K)	x 1.000.000 (M)
1 Ω	10 Ω	100 Ω	1 KΩ	10 KΩ	100 KΩ	1 MΩ
1,2 Ω	12 Ω	120 Ω	1K2 Ω	12 KΩ	120 KΩ	1M2 Ω
1,5 Ω	15 Ω	150 Ω	1K5 Ω	15 KΩ	150 KΩ	1M5 Ω
1,8 Ω	18 Ω	180 Ω	1K8 Ω	18 KΩ	180 KΩ	1M8 Ω
2,2 Ω	22 Ω	220 Ω	2K2 Ω	22 KΩ	220 KΩ	2M2 Ω
2,7 Ω	27 Ω	270 Ω	2K7 Ω	27 KΩ	270 KΩ	2M7 Ω
3,3 Ω	33 Ω	330 Ω	3K3 Ω	33 KΩ	330 KΩ	3M3 Ω
3,9 Ω	39 Ω	390 Ω	3K9 Ω	39 KΩ	390 KΩ	3M9 Ω
4,7 Ω	47 Ω	470 Ω	4K7 Ω	47 KΩ	470 KΩ	4M7 Ω
5,1 Ω	51 Ω	510 Ω	5K1 Ω	51 KΩ	510 KΩ	5M1 Ω
5,6 Ω	56 Ω	560 Ω	5K6 Ω	56 KΩ	560 KΩ	5M6 Ω
6,8 Ω	68 Ω	680 Ω	6K8 Ω	68 KΩ	680 KΩ	6M8 Ω
8,2 Ω	82 Ω	820 Ω	8K2 Ω	82 KΩ	820 KΩ	8M2 Ω
						10M Ω

82 ohms es la ideal, pero aún así usaremos una resistencia de 100 ohms.

La mayoría de las aplicaciones depende de una batería y si limitamos el uso de la corriente eléctrica podemos aumentar la vida del producto, esto es hacer que el consumidor final no tenga que reemplazar las baterías muy seguido.

Selección de la Resistencia

Con una resistencia de 100 ohms, ¿Cuánta corriente fluirá por el Led?

$R = 13 \text{ mA}$.

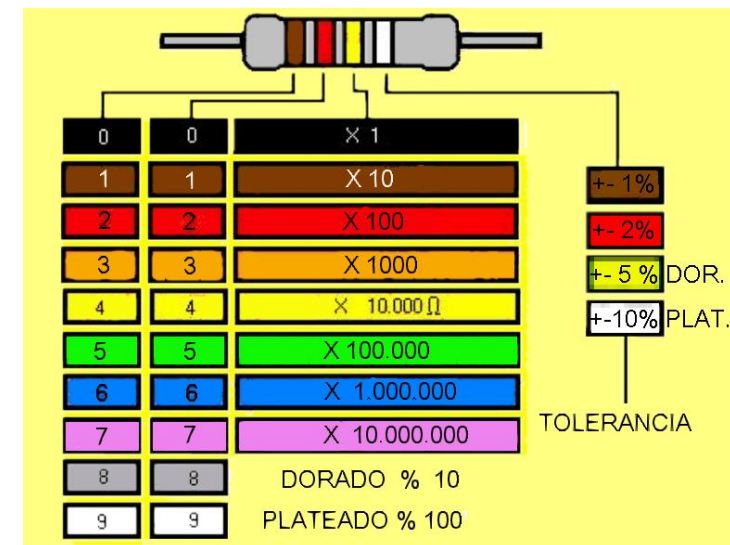
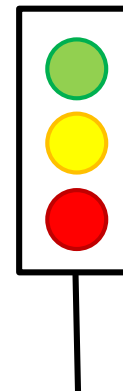
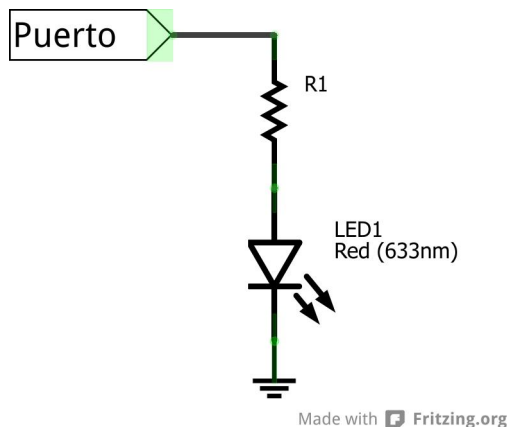
Semáforo con Leds

En esta práctica armaremos y programaremos un semáforo que siga la siguiente secuencia:

1. Verde por 3 segundos
2. Parpadea verde 3 veces (500 ms encendido, 500 ms apagado)
3. Prende amarillo por un segundo
4. Prende rojo por 3 segundos

Recuerda que el semáforo no debe de estar apagado en ningún momento (excepto cuando el verde está parpadeando).

No hay 2 luces prendidas al mismo tiempo.



Práctica 6

Semáforo paso peatonal

Entradas Digitales

En las prácticas anteriores vimos como utilizar los puertos como salidas digitales las cuales pueden proveer voltaje (“1”) o tierra (“0”).

Cuando un puerto está configurado como entrada digital este es capaz de visualizar dos estados, “0” o “1”

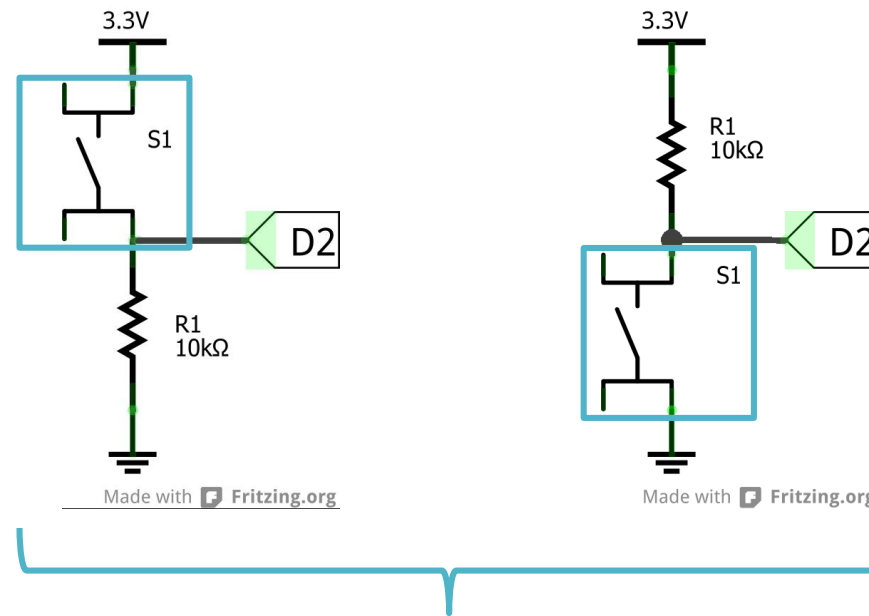
¿Voltaje de umbral?

Voltaje en el cual el microcontrolador distingue entre un “1” o un “0” lógico

Botón Normalmente Abierto

El siguiente diagrama muestra como se conecta un botón.

Ejemplo de código de como leer una entrada digital



¿Qué diferencia hay?

Análisis del Circuito

Se analizarán los 2 circuitos anteriores

Resistencias Internas

El microcontrolador puede poner resistencia de PullUp en sus puertos cuando estos están como entradas.

[Ver código](#)

Paso Peatonal

Un semáforo para vehículos deberá estar siempre en verde.

Si una persona presiona un botón el semáforo de vehículos hace transición a rojo de la siguiente manera:

1. *Parpadea verde 3 veces (a 1 Hz)*
2. *Amarillo por un segundo*
3. *Rojo por 3 segundos*
4. *Verde a esperar que el botón sea presionado de nuevo.*

Clase 3

1. Entradas Analógicas
2. Comunicación Serie (por USB)

KL25Z

Contenido del día

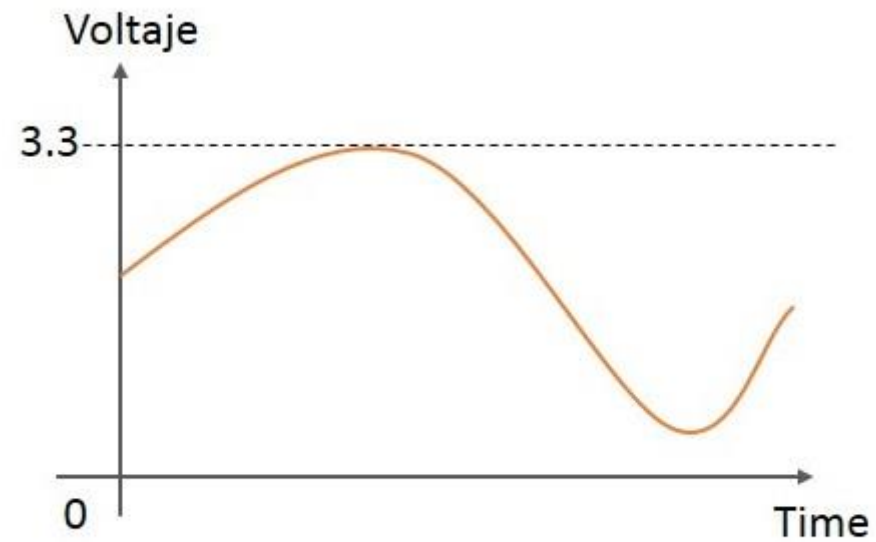
Entradas Analógicas

¿Qué es una señal analógica?

Da ejemplos de señales analógicas

Señal Analógica

Analog signal



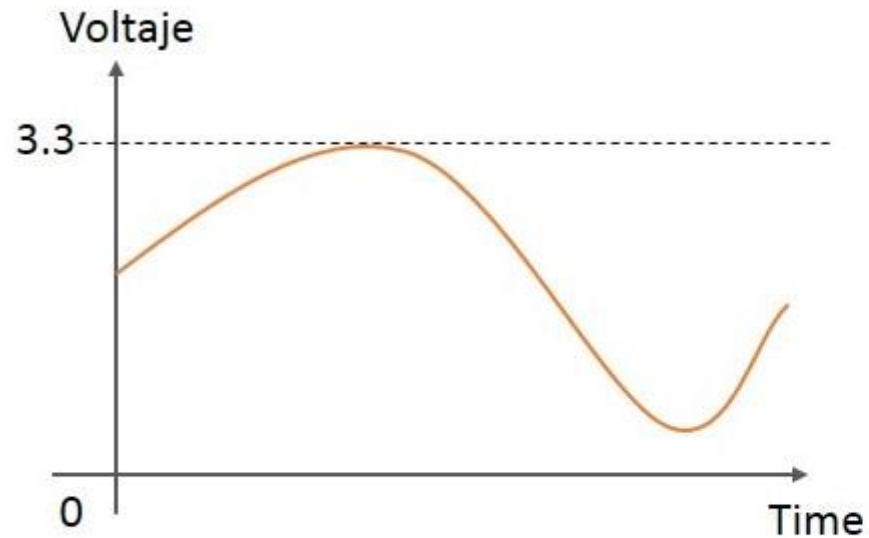
Has a determined value in each time instant

Ejemplos de Señales Analógicas

- Vos
- Señal de un radar
- FM
- Señal neurovascular

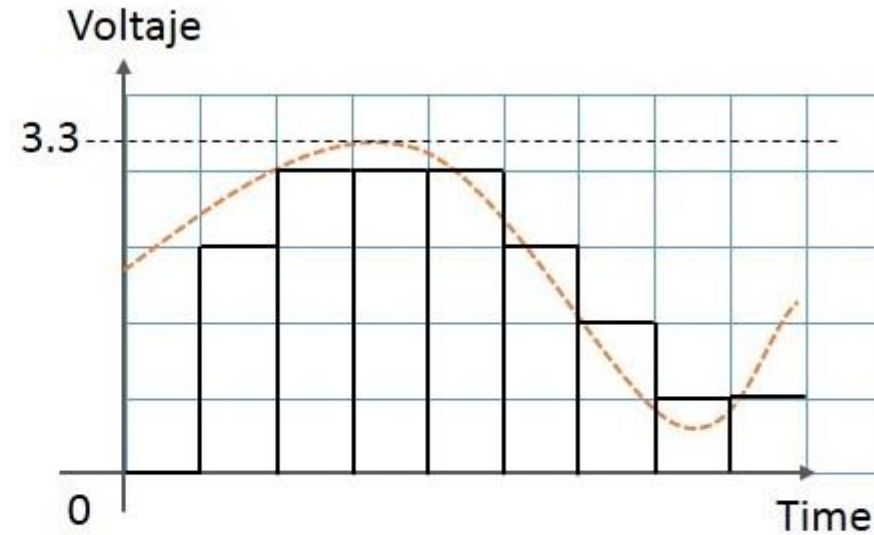
Señales Analógicas vs Digitales Discretas

Analog signal



Has a determined value in each time instant

Digital signal



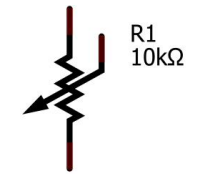
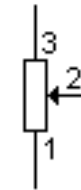
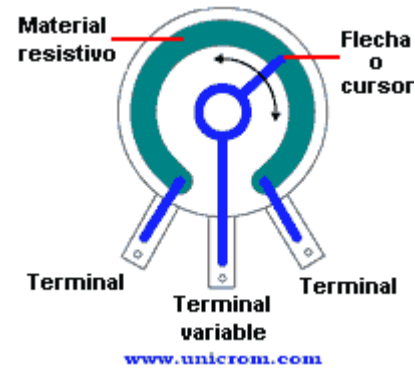
It refresh values each time a sample is taken. It rounds the value to the nearest group of possible values.

Potenciómetros

Son resistencias variables

Tienen un valor nominal

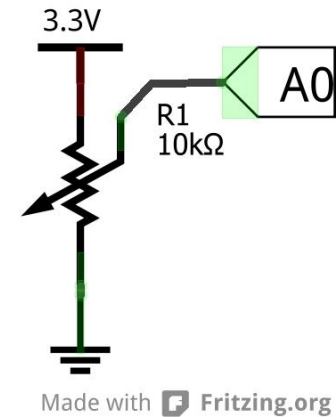
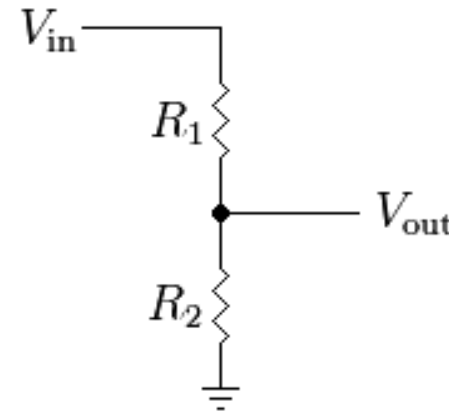
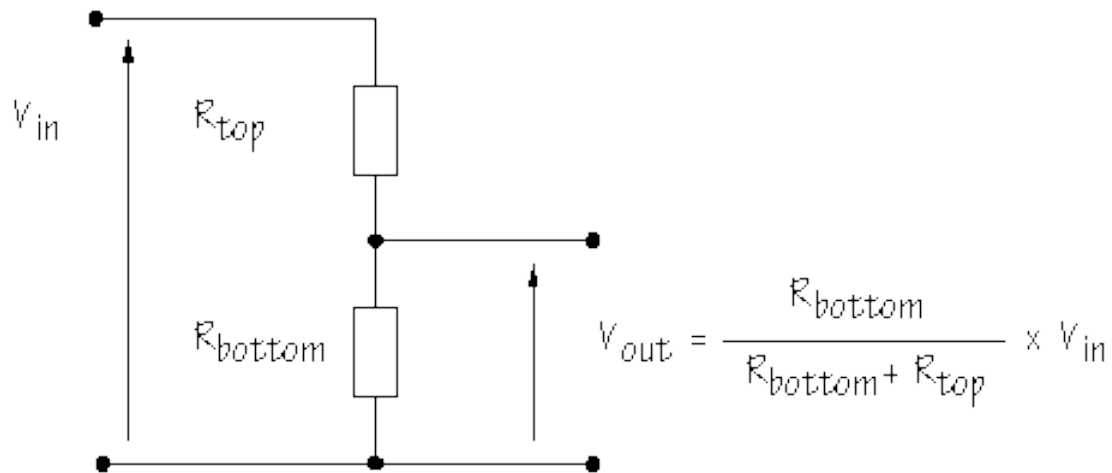
Se utilizan en diversas aplicaciones como controles de volumen.



with  Fritzing.org

Divisor de Voltaje

Un divisor de voltaje está constituido por 2 resistencias en serie que dividen el voltaje.



Para que sirve el divisor de voltaje

- Es útil cuando por lo menos una resistencia es variable. Hay sensores que cambian su resistencia dependiendo de alguna variable, dos ejemplos:
 - Fotorresistencia: Varía su resistencia dependiendo de la cantidad de luz visible que recibe.
 - Termistor: Varía su resistencia dependiendo de la temperatura.

Potenciómetro como Divisor de Voltaje

Utilizaremos el potenciómetro para controlar el comportamiento de Leds. Como lo verás en las siguientes prácticas

Práctica 7

Controla la intensidad de un led con un potenciómetro

Práctica 8

Dos leds parpadean intermitentemente a una frecuencia dada por el potenciómetro. Rango de 50 a 500 ms (periodo)

Práctica 9

Medir la intensidad de una señal analógica y mostrarla en 4 leds (verde, verde, amarillo, rojo)

Práctica 10

Comunicación Serie con la computadora

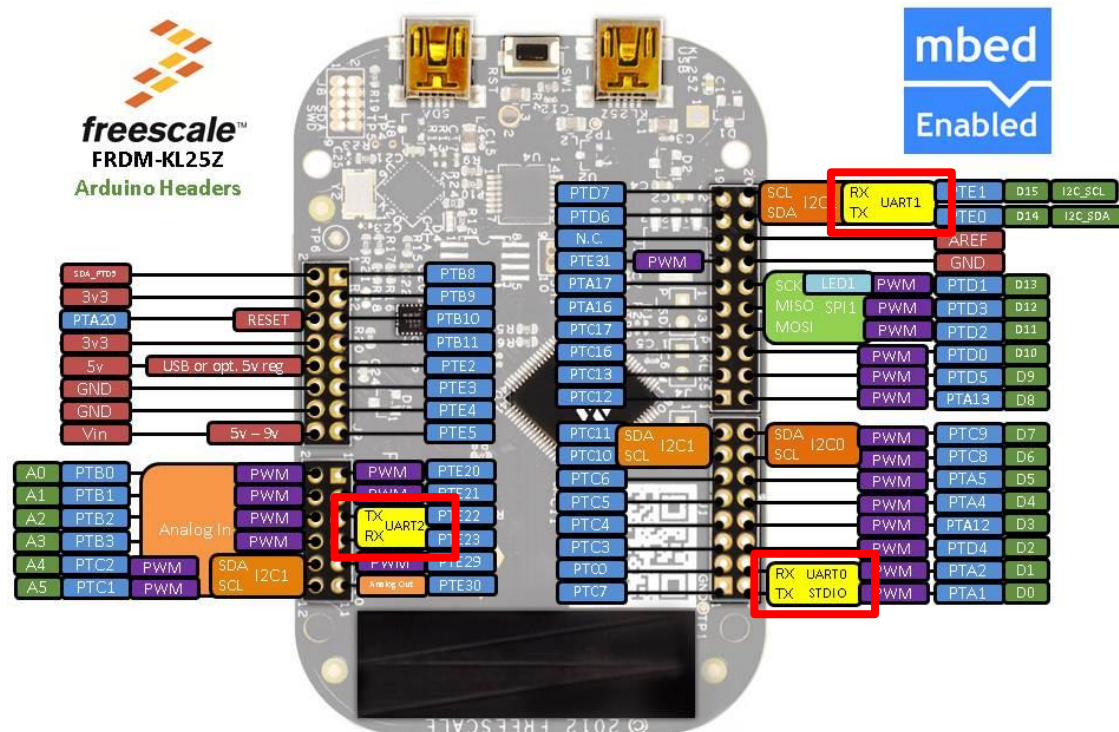
Comunicación Serie

Utilizaremos la comunicación Serie para mandar datos desde la tarjeta Freedom hacia la computadora por medio del cable USB. Para poder hacer esto necesitamos:

1. Programar nuestra tarjeta para que pueda mandar datos hacia la computadora
2. Configurar el driver en nuestra computadora
3. Descargar un programa para poder visualizar datos y configurarlo.

Programación

Necesitamos crear un objeto de la clase Serial y poner los pines por donde se realizará la comunicación Serie. Hay tres pares de pines serie que podemos utilizar, en este caso utilizaremos el PTA1 y PTA2 por que son los que están conectados al puerto USB.



Programación

Realiza la programación necesaria para leer un potenciómetro conectado como divisor de voltaje y mande el dato leído a través del puerto Serie.

Manda por el puerto Serie el texto que se encuentre entre "", %f es para indicar que debe colocar un valor tipo float en esa posición, \n es para dar un salto de línea. Potval es la variable tipo float que va a poner después de es: y antes de volts.

```
#include "mbed.h"
Serial pc(USBTX, USBRX); // tx, rx
AnalogIn pot(A0);

float potval;

int main(){
    while(1){
        potval = pot;
        pc.printf("El valor del puerto es: %f volts\n", potval);
        wait(1);
    }
}
```

Se crea el objeto, en este caso se llama pc y se le asignan dos puertos uno de transmisión y otro de recepción de datos.

Driver

Seguir las instrucciones de: <http://mbed.org/handbook/Windows-serial-configuration>

Tera Term - Windows

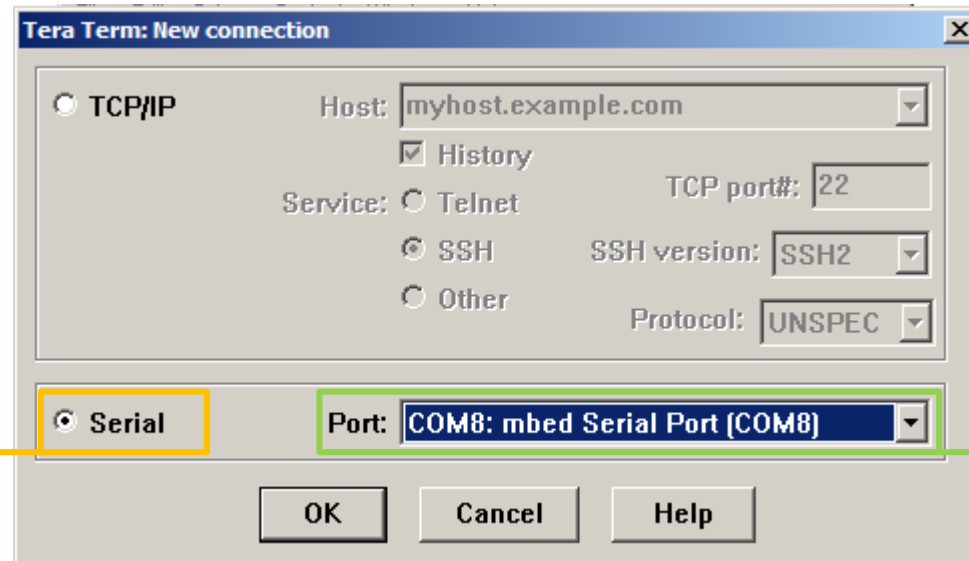
Tera Term es un programa que entre varias características puede monitorear los datos que pasan a través de un puerto COM (USB).

Para instalarlo sigue los siguientes pasos:

1. *Buscar en google Tera Term*
2. *Revisar cual es la última versión (4.79)*
3. *Seguir instrucciones de descarga*
4. *Ejecutar el archivo .exe*
5. *Seguir los pasos del instalador*
 - a. *Instalar la versión "Estándar" o "Full"*

Configuración

Ejecuta Tera Term, aparecerá una pantalla como la que se muestra a continuación

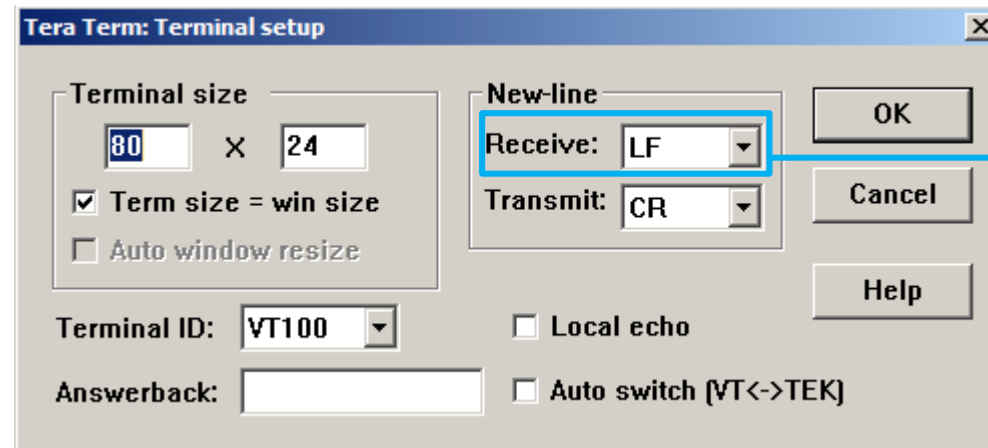


Selecciona Serial

De la lista que aparece busca el dispositivo con el nombre mbed

Configuración

Dentro del menú “*Setup >> Terminal*” cambiar la configuración para recibir línea nueva:



Cambia la opción a “LF”

Otras Plataformas

Mac OS X - Use the command `ls /dev/tty.usbmodem*`

Linux

Primero ve la lista de dispositivos usb con el comando:

```
ls /dev/ttyACM*
```

Después utiliza el siguiente comando, sustituyendo por el puerto donde está conectado tu plataforma:

```
screen /dev/ttyACM0
```

Clase 4

1. Acelerómetro 3 ejes

- a. Características
- b. Mandando datos a la computadora

KL25Z

Contenido del día

Acelerómetro MMA8451Q

- 3 ejes (X, Y y Z).
- Medición máxima de 3g en los ejes X y Y y de 5g en Z.
- 14 bits: Describe las aceleraciones con una precisión de 4096.
- Comunicación I2C

Librerías

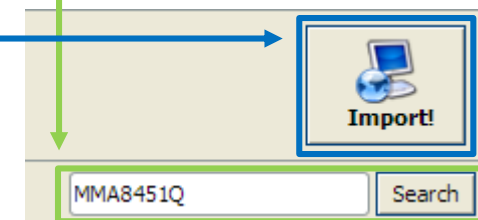
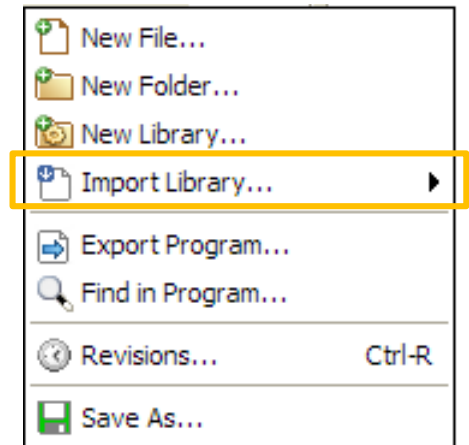
Para poder utilizar el acelerómetro que viene integrado en la tarjeta de desarrollo (MMA8451Q) tenemos que importar una librería que se encarga de dar las configuraciones iniciales y establecer comunicación con el dispositivo haciendo más fácil la integración con el microcontrolador.

Con la librería podemos pedir la medición de la aceleración en cualquier eje con solo una línea de código.

Importar una librería

Para importar una librería sigue los siguientes pasos:

- Clic derecho sobre el programa al que quieres importar una librería.
- Selecciona *Import Library...*
- Clic en *From Import Wizard...*
- Se abrirá una ventana en la cual podemos buscar la librería
- En el cuadro de búsqueda escribe “MMA8451Q”
- Selecciona la librería escrita por “Johan Kritzinger”
- Clic en *Import!*



Práctica 11

Acelerómetro

Ejemplo

El siguiente programa lee el acelerómetro en los 3 ejes y prende el led RGB de acuerdo a cada una de las aceleraciones.

Aceleración	Color
Eje X	Rojo
Eje Y	Verde
Eje Z	Azul

Práctica 12

Agregar comunicación Serie

Clase 5

1. Sensor Capacitivo
2. Motores Servo
 - a. Servo Estándar
 - b. Servo de Rotación Continua
3. Proyecto Integrador
4. Examen Segundo Parcial

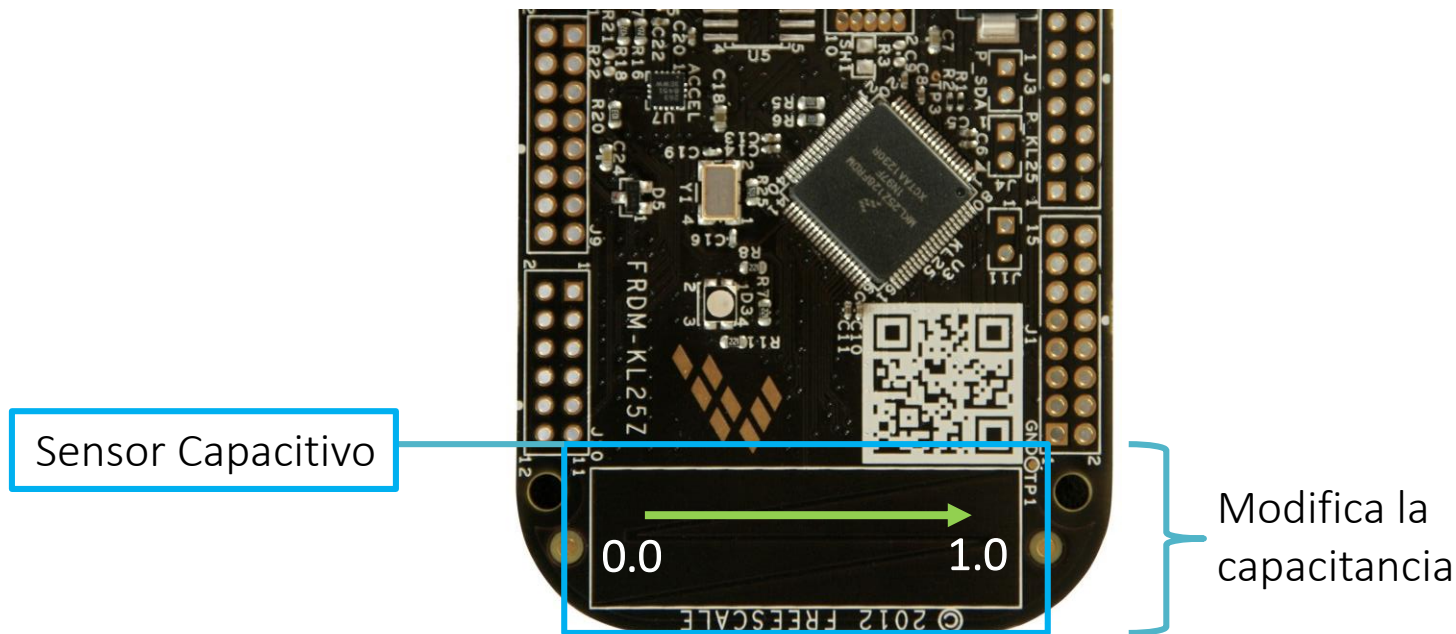
KL25Z

Contenido del día

Sensor Capacitivo

Sensor Capacitivo

La tarjeta de desarrollo cuenta con un sensor capacitivo integrado, el cual permite reconocer en que lugar del sensor se ha colocado el dedo.



Código Sensor Capacitivo

El siguiente código muestra como controlar una señal PWM (LED1) con el sensor capacitivo.

```
#include "mbed.h"
#include "TSISensor.h"

PwmOut led(LED_GREEN);
TSISensor tsi;

int main(void) {
    led = 1;
    while (true) {
        led = 1.0 - tsi.readPercentage();
        wait(0.1);
    }
}
```

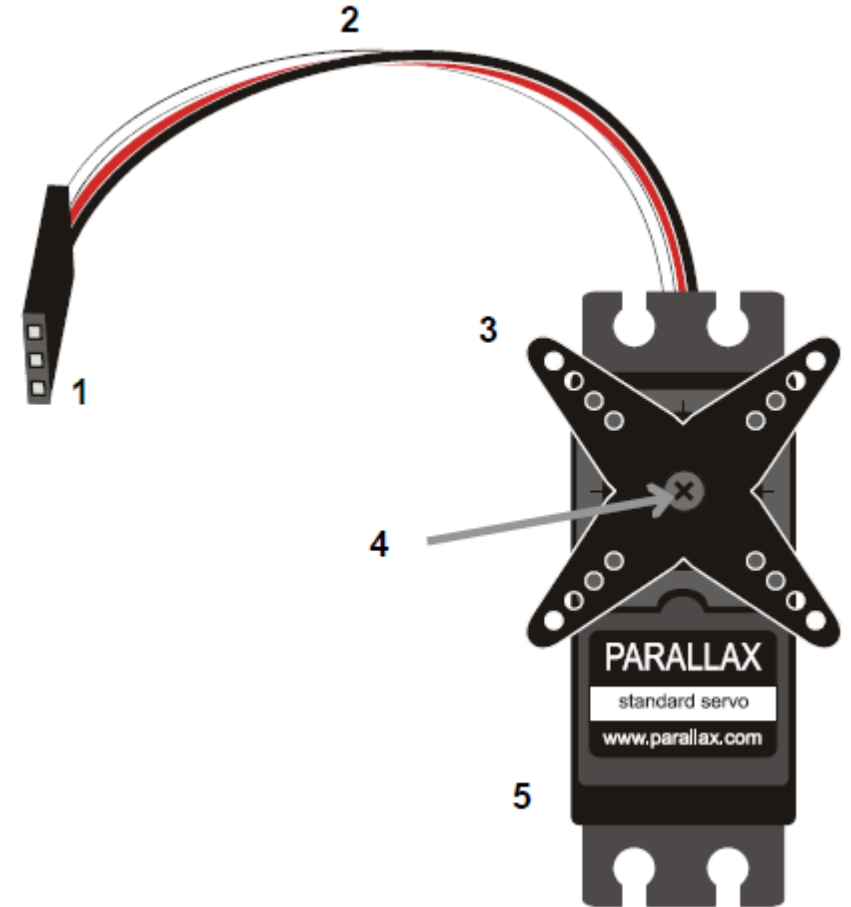
Motores Servo

Motores Servo

En la figura se muestra un motor servo.

Los números representan:

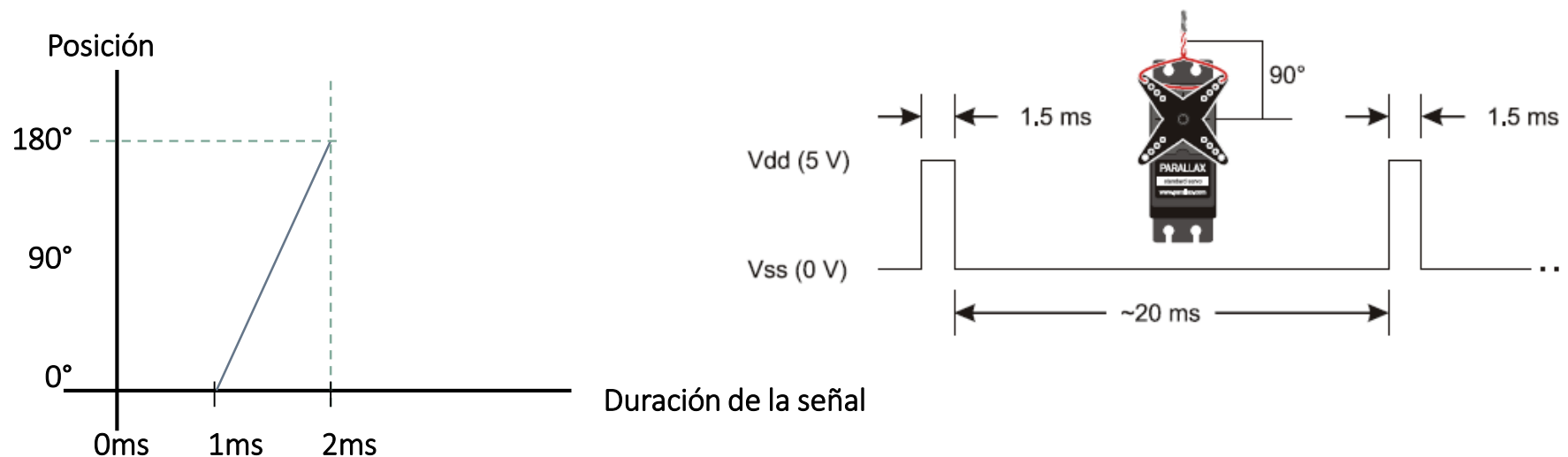
1. Conector
2. Cables: Rojo (Vdd), Negro (Vss) y Blanco (Señal)
3. Cruz de plastico
4. Tornillo que sostiene a la cruz
5. Cuerpo del motor servo



¿Cómo Funciona un Servo?

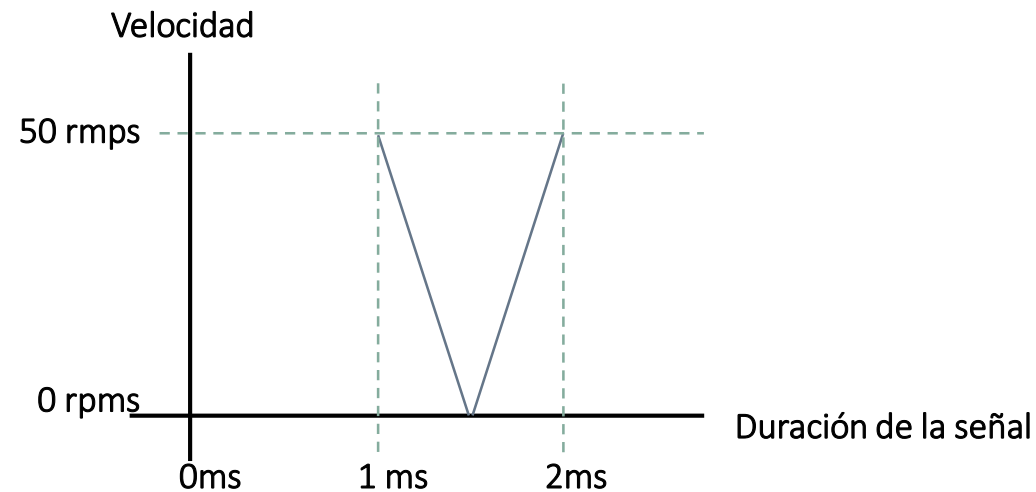
Por medio de cable de Señal (Blanco) se manda una señal (un “1” lógico) durante determinado tiempo, dependiendo de la duración de esa señal, es la posición a la que se mueve el motor.

Normalmente los motores servo solo giran 180° , pero hay otros que giran solo 90° o 270° , entre otros.



Servo de Rotación Continua

Los servos que están montados en el robot giran continuamente para poder desplazarse. Se controlan de la misma forma, pero la respuesta es diferente, no se controla posición, sino velocidad.



Clase 6

1. Tipos de Funciones
2. Emular un Mouse USB

KL25Z

Contenido del día

Tipos de Funciones

Tipos de Funciones

Existen 4 tipos de funciones

T1 No requieren de valores y no regresan un valor, solo ejecutan código

T2 Requieren de uno o mas valores y no regresan un valor

T3 No requiere de valores, pero si regresan un valor

T4 Requieren de uno o varios valores y regresan un valor

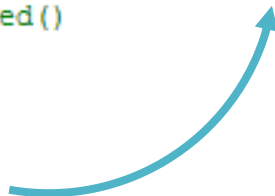
Tipo 1

No requieren de valores y no regresan un valor, solo ejecutan código.

Ejemplo:

```
7 float t = 0.5;
8
9 // Todas las funciones se tienen que declarar al inicio
10 void red();
11 void green();
12 void blue();
13 void pink();
14 void cyan();
15 void yellow();
16 void white();
17 void colors_off();
18
19 int main() {
20     while(1) {
21         red(); // Manda a llamar la funcion red()
22         wait(t);
23         blue();    wait(t);
24         green();  wait(t);
25         cyan();   wait(t);
```

```
26         yellow();    wait(t);
27         pink();       wait(t);
28         white();     wait(t);
29         colors_off(); wait(t);
30     }
31 }
32
33
34 void red(){ // Código de la funcion red()
35     rled = 0;
36     gled = 1;
37     bled = 1;
38 }
```



Tipo 2

Requieren de uno o mas valores y no regresan un valor

```
1 #include "mbed.h"
2
3 PwmOut bled(LED1);
4 PwmOut gled(LED2);
5 PwmOut rled(LED3);
6
7 void color(int red, int green, int blue);
8
9 int main() {
10     color(20,20,255);
11     while(1) {
12
13     }
14 }
15
```

```
16 void color(int red, int green, int blue){
17     float r, g, b;
18
19     r = (float)red/255;
20     g = (float)green/255;
21     b = (float)blue/255;
22
23     bled = 1 - b;
24     gled = 1 - g;
25     rled = 1 - r;
26 }
```




Tipo 3

No requiere de valores, pero si regresan un valor

```
1 #include "mbed.h"
2
3 DigitalOut myled(LED1);
4 AnalogIn sensorTemperatura(A0);
5
6 float temperatura;
7
8 float calcular_temperatura();
9
10 int main() {
11     while(1) {
12         temperatura = calcular_temperatura();
13         if (temperatura > 20){
14             myled = 0;
15         }else{
16             myled = 1;
17         }
18     }
19 }
```

```
20
21
22 float calcular_temperatura(){
23     float miliVolts, temp;
24     miliVolts = (sensorTemperatura * 3300)/1;
25
26     temp = miliVolts / 10;
27
28     return temp;
29 }
```



Tipo 4

Requieren de uno o varios valores y regresan un valor

Emular un Mouse USB

Mouse

Funciones	Descripción
<code>move(x,y)</code>	Mueve el mouse “x” y “y” pixeles, enteros positivos y negativos
<code>scroll(z)</code>	Scroll, enteros positivos y negativos
<code>doubleClick()</code>	Doble clic del botón izquierdo
<code>click(button)</code>	Clic de botón izquierdo (MOUSE_LEFT) o derecho (MOUSE_RIGHT)
<code>press(button)</code>	Presionar botón izquierdo (MOUSE_LEFT) o derecho (MOUSE_RIGHT)
<code>release(button)</code>	Soltar botón izquierdo (MOUSE_LEFT) o derecho (MOUSE_RIGHT)

Clase 7

Sensores

Piezo Film Sensor

APPLICATIONS:

- Vibration Sensing in Washing Machine
- Low Power Wakeup Switch
- Low Cost Vibration Sensing
- Car Alarms
- Body Movement
- Security Systems Alarm system sensor
- Product damage/shock detector
- Acceleration sensor
- Tap switch

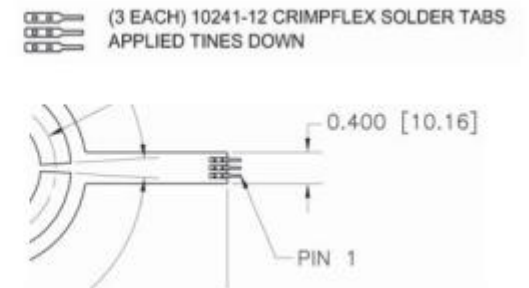
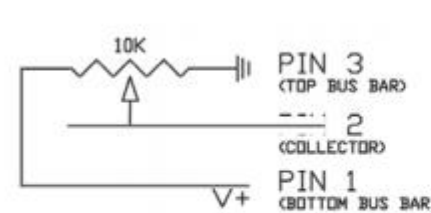
Potenciómetro Softpot en Circulo

FUNCIONAMIENTO:

- Resistencia entre 100 Ohms a 10,000 Ohms
- Resistente a líquidos, puede operar en ambientes médicos.
- No se debe presionar en la parte donde sale el conector o se calentará y eventualmente se quemará.

APLICACIONES:

- Control de volumen, brillo, temperatura, etc.

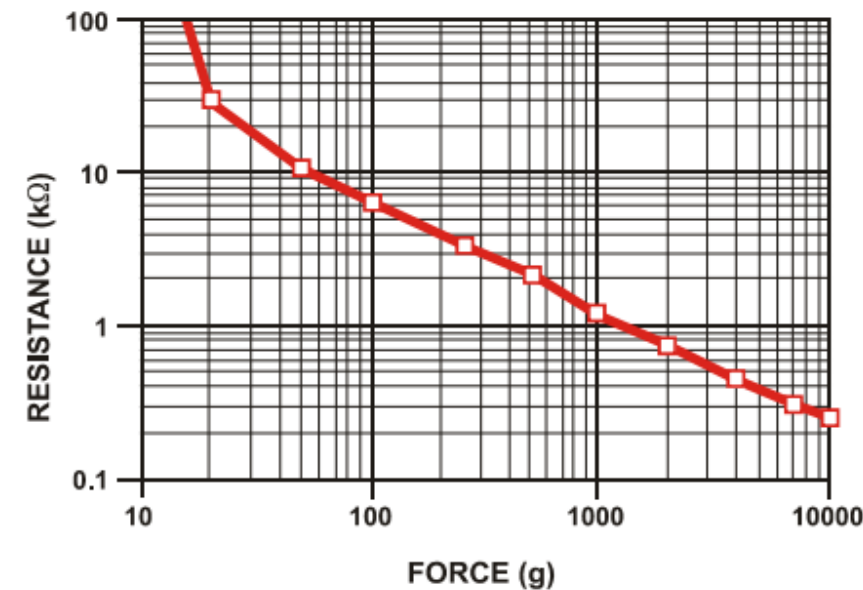


Force Sensitive Resistor

APLICACIONES:

- **Detect & qualify press:** Sense whether a touch is accidental or intended by reading force
- **Use force for UI feedback:** Detect more or less user force to make a more intuitive interface
- **Enhance tool safety:** Differentiate a grip from a touch as a safety lock
- **Find centroid of force:** Use multiple sensors to determine centroid of force.
- **Detect presence, position, or motion:** Of a person or patient in a bed, chair, or medical device
- **Detect liquid blockage:** Detect tube or pump occlusion or blockage by measuring back pressure
- **Detect proper tube positioning**
- **Many other force measurement applications**

Sensibilidad: 0.1N a 1.0N



Reed Switch

FUNCIONAMIENTO:

- Switch activado por un imán.



APLICACIONES:

- Detectar cuando algo se abre/cierra (puerta, ventana, tapa, etc)
- Contar revoluciones o vueltas*
- Saber la posición de algún actuador
- Saber si algo contiene todos los elementos.

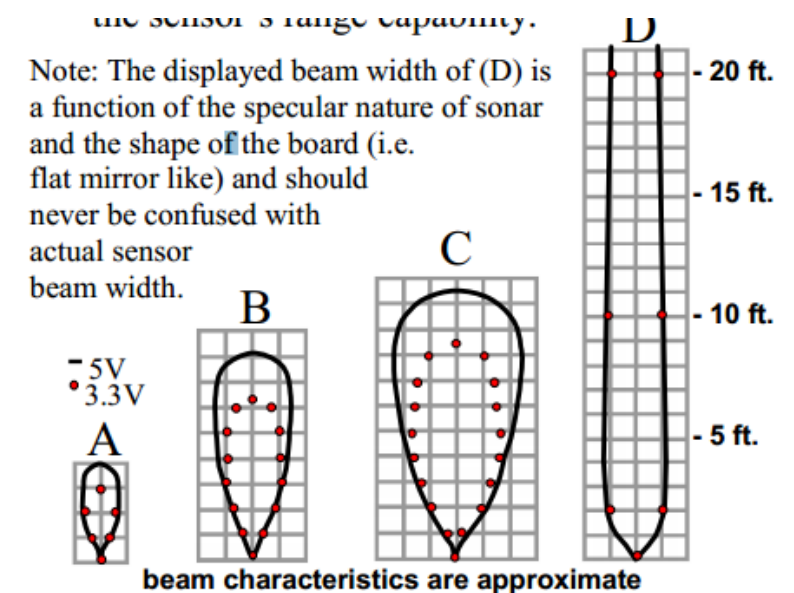
Sonar Range Finder (Ultrasonico)

CARACTERÍSTICAS:

- Detecta objetos entre 0 y 645 cms.
- Mide distancia entre 15 y 645 cms.
- Muchas formas de conectar a un microcontrolador.
- Uso en interiores

APLICACIONES:

Medir distancia



Microfono Electret

FUNCIONAMIENTO:

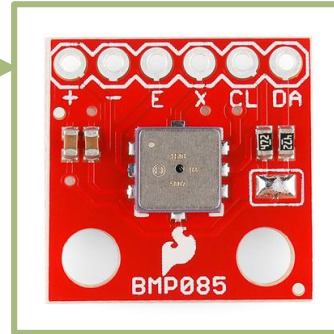
- Micrófono con amplificador integrado.
- Salida analógica.

APLICACIONES:

- Detectar ruido.
- Detectar secuencia de sonido*

Otros Sensores

- Fotorresistencias (Luz)
- Sensor Sharp (Distancia)
- Sensor de color
- Potenciómetros deslizables
- Presión
- Temperatura
- Otros

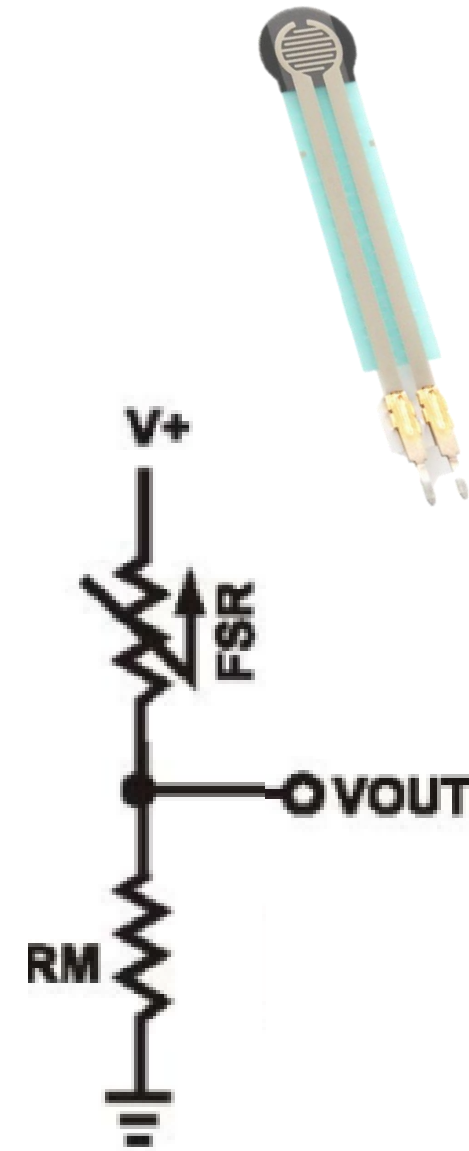


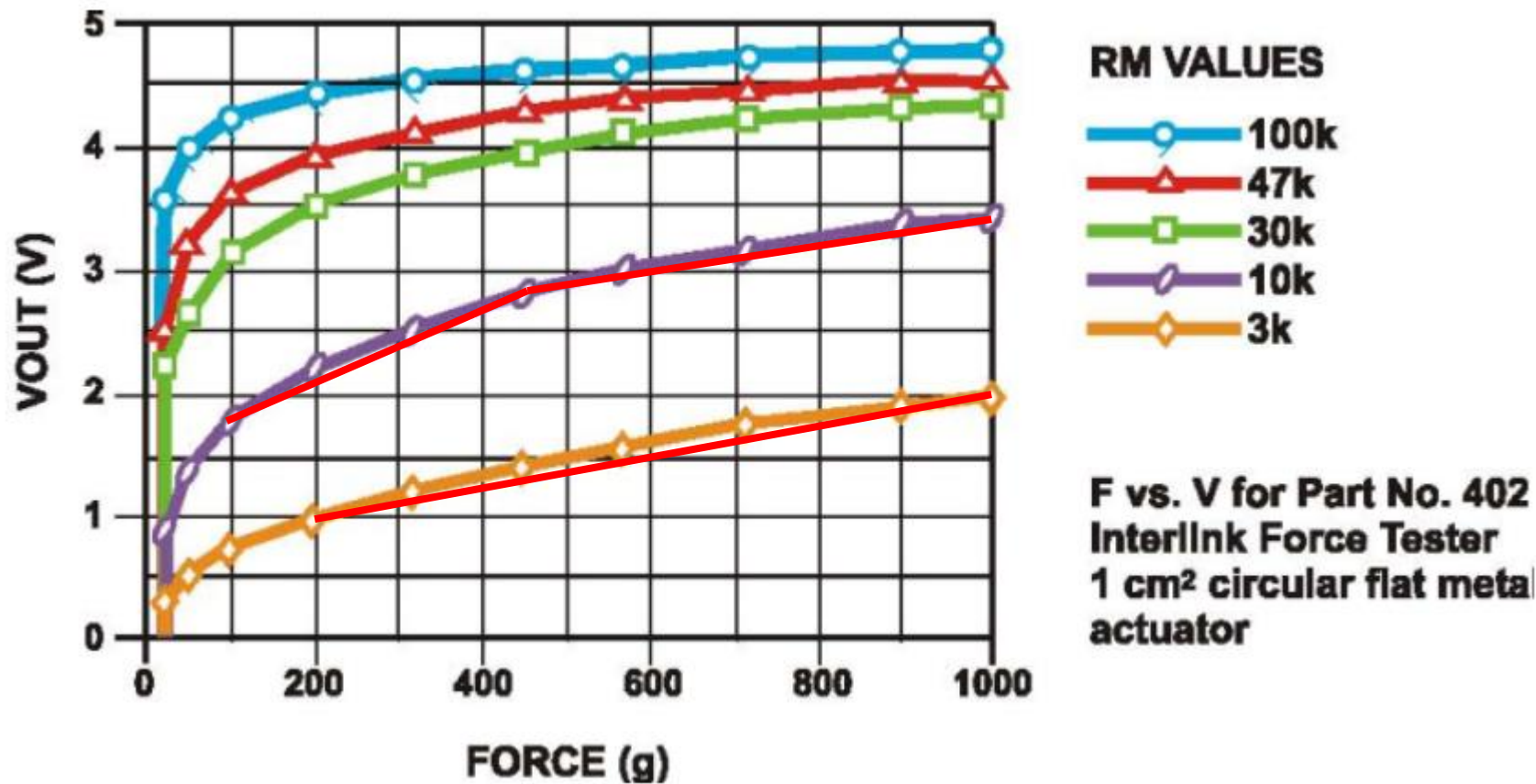
Sensor de Fuerza

FSR 400 – Sensor de Fuerza

Rango de medición: 0.1 - 10.0 Newtons

Number of Actuations (Life time): 10 Million tested, without failure.





Asumiendo V+ como 5V

$$V_{OUT} = \frac{R_M V_+}{(R_M + R_{FSR})}$$

Licencia de uso

Utiliza libremente este documento siempre respetando los derechos de autor.

Si encuentras errores mándame las correcciones a mi correo.

Puedes contribuir a enriquecer la presentación mandándome una copia a mi correo.

Dudas / Preguntas / comentarios: gcarmonar@gmail.com

Autor: Gerardo Carmona, makerobots.tk