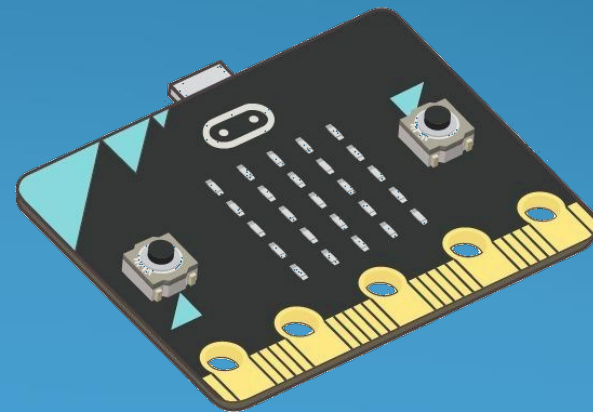
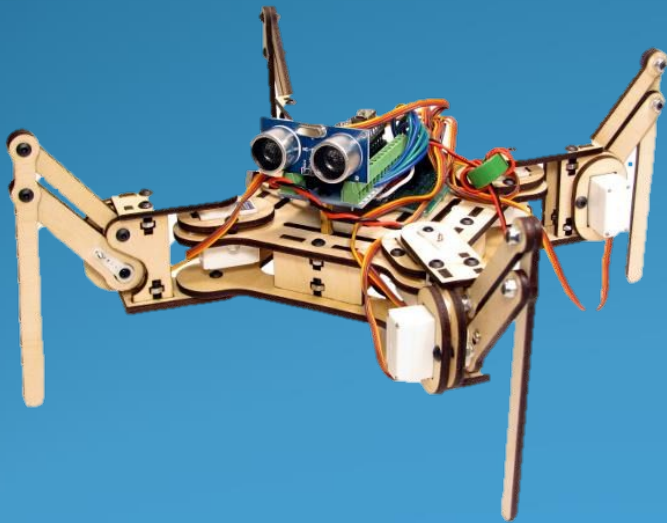


# Tallerine Biónico

2024

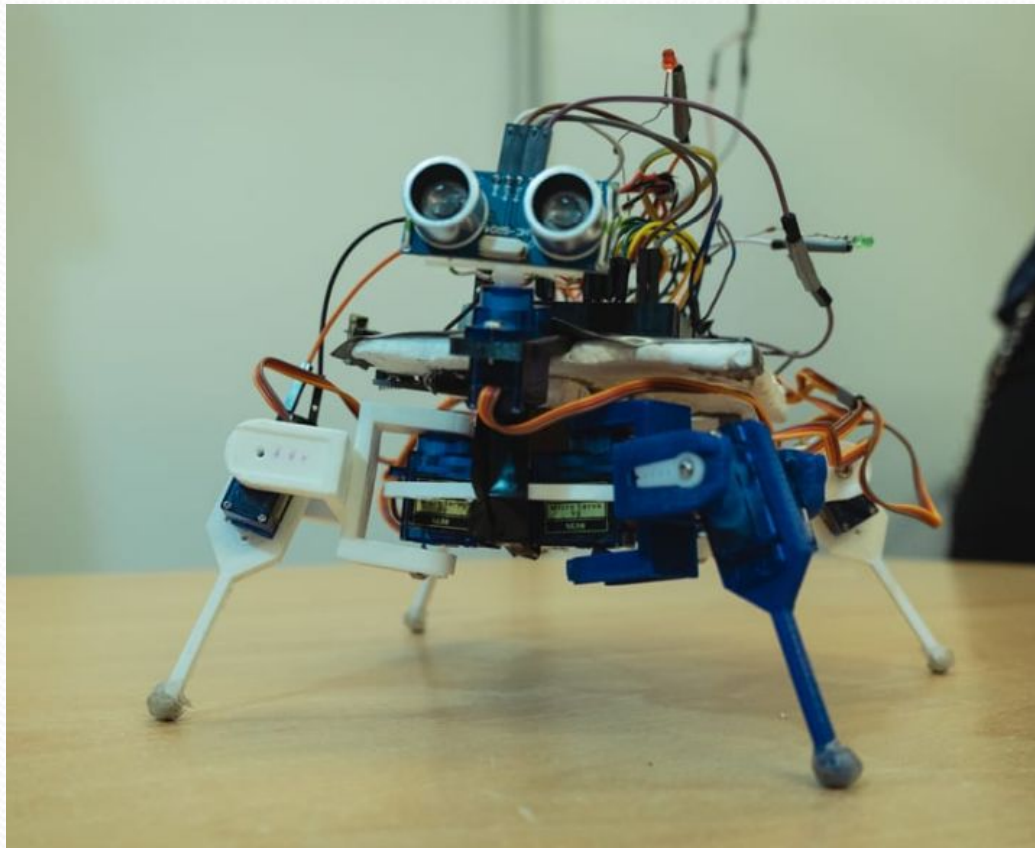


Taller de Introducción a la Ingeniería Eléctrica  
Instituto de Ingeniería Eléctrica - Facultad de Ingeniería – Universidad de la República

# Objetivo

- Actividad orientada a la generación de ingreso
- Introducción a la Ingeniería Eléctrica
- Motivación
- Creatividad
- Trabajo en equipo
- Enseñanza activa:
  - Se viene a **HACER**, no solo escuchar
- El medio para lograr lo anterior:
  - Construir un “bicho” que camine y mucho más...

# Les presentamos al “bicho”



# Equipo docente

## ● Docentes:

- Javier Rodríguez (responsable)
- Leandro Díaz
- Matías García

## ● Ayudantes:

- Mateo Díaz
- Rodrigo Fernández

# Clases y carga horaria

- Clases teórico/prácticas:
  - Día y hora: Miércoles de 16:30 a 19:30
  - Lugar: Laboratorio de Medidas IIE
- Clases de consulta
  - **A demanda**..... (Se registra asistencia, del grupo y estudiantes)
- Carga horaria:
  - 10 horas semanales durante 15 semanas. Total: 10 créditos
    - 3 horas semanales en clase
    - **7 horas semanales de trabajo por estudiante!!!**

# Comunicación oficial del curso

- Herramienta EVA:
  - Sitio: **<https://eva.fing.edu.uy>**
    - Allí seleccionar: Institutos > Ingeniería Eléctrica > Cursos del Primer semestre 2024 > Tallerine > Tallerine Bionico
  - Dentro del EVA hay:
    - Foros de información
    - Foros de consulta
    - Material teórico
    - Material de apoyo
    - Ejercicios a realizar
    - .....

# Organización del curso

- Presentación inicial
- Obligatorios semanales. Trabajo guiado.
  - Armado del bicho. Funciones básicas (adelante, atrás, derecha, izquierda), Bluetooth
- Trabajo libre
  - Integración completa (ensayo y error). Aplicación final. Personaje, con un “nombre”. Diseño artístico. **Mucha creatividad.**
- Preinforme e Informe
- Presentación final
- Realización de un poster
- Realización de un video
- Presentación en Muestra de todo Tallere (sem 05/08)

# Asistencia

- La asistencia a clases es **OBLIGATORIA** (se pasa lista)
- Se puede faltar **solo 2 veces**.
- Las clases son de 3 horas. Tolerancia excepcional 15 min a la entrada y a la salida.
- La muestra final de agosto es **obligatoria**.  
Fecha aprox. 07/08 de 9 a 12 hs



# Trabajo en grupos

- **Los grupos son de 5 estudiantes**
- Los grupos se arman ahora!!!!

# Evaluación

La evaluación consta de:

- Concepto individual
- Evaluación de obligatorios. Sin terminar **RESTA**
- Proyecto: solución final, armado, parte artística.
- Preinforme e informe
- Presentación final y dispositivas.
- Poster y video.
- Muestra tallerine (asistencia obligatoria).
- Devolución del material (es condición para aprobar el curso).

# Evaluación

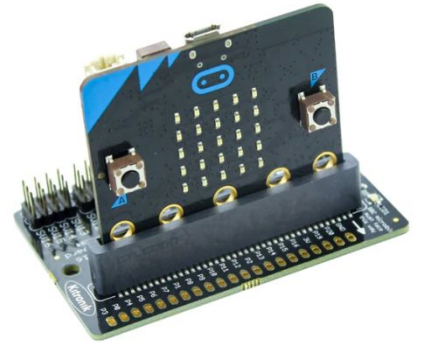
Algunas aclaraciones:

- Concepto individual:
  - Evaluación continua.
  - Trabajo en grupo.
  - Asistencia (llegar tarde, retirarse antes).
  - Co-evaluación
  - Se toma muy en cuenta la **ACTITUD**.
    - El que no hace nada  MUY MAL
    - El que lo hace todo  **MUCHO PEOR**
  - Evaluación individual sobre los obligatorios en forma oral.

# Material

## ● Material a recibir por grupo:

- Placa micro:bit
- Placa controlador de servos Kitronik
- 9 servomotores
- 1 sensor de ultrasonido
- Partes para armar un cuadrúpedo (4 patas, 9 articulaciones, 2 caderas, 1 columna, tornillos).
- Otros .. ver la lista de materiales entregada con el kit



# Material

Material	Grupo X
Micro:bit	1
Servo driver	1
Cable USB corto microbit	1
Servomotores	11
Aspas de una paleta	10
Tornillos para aspas	12
Aspas de dos paletas	2
Aspas de cuatro paletas	2
Patas	4
Articulaciones	9
Caderas	2
Columna	1
Tornillos con tuerca	10
Tornillos para servos	27
Rollo de cable para atar	1
Protoboard	1
Sensor ultrasonido	1
Cable USB para alimentación	1
Led + resistencia 330R	3
Cables macho/macho	5
Cables macho/hembra	10
Cable hembra/hembra	5

# Material

## ● Entrega del material

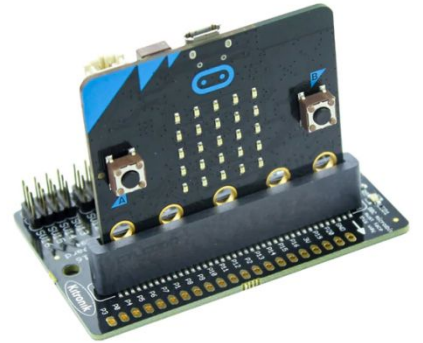
- Se entrega el primer día de clase.
- Se debe revisar todo el material
- Una vez aceptado, no hay reclamos posteriores.

## ● Devolución del material

- NO participan en IDM: semana del 12/08
- SI participan en IDM: semana siguiente a finalizar IDM.
- Se coordina para revisar que todo esté en orden.
- Se debe REPONER lo que falte y lo dañado.

## Referencia

- El material completo cuesta US\$ 300.
- Como mínimo, cada grupo romperá 4 servos.



# Material

## ● Material no proporcionado:

- Cuerpo
- Soporte para cabeza
- Vestimenta (creatividad)
- Fuente de 5V, 2A (cargador de celular con salida USB)

## ● Material opcional no proporcionado

- Power Bank ( 5V, mínimo 2.1 A)

# Ingeniería de Muestra (IdM)

- Evento anual que organiza la Facultad de Ingeniería donde se presentan proyectos, investigaciones y emprendimientos de estudiantes y docentes, con el fin de exhibir las actividades que se realizan en la Institución.
- Participación voluntaria.
- Tallere Biónico participa siempre!!!



*IdM Salto 2018*



*IdM Paysandú 2019*

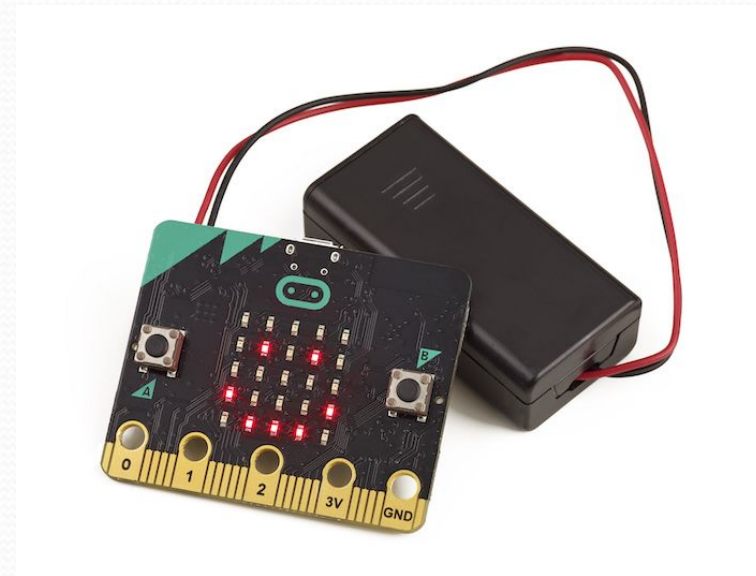




# VIDEOS!!!

# Micro:bit

- ¿Qué es Micro:bit?
- ¿Un microcontrolador?
- ¿Para qué sirve?



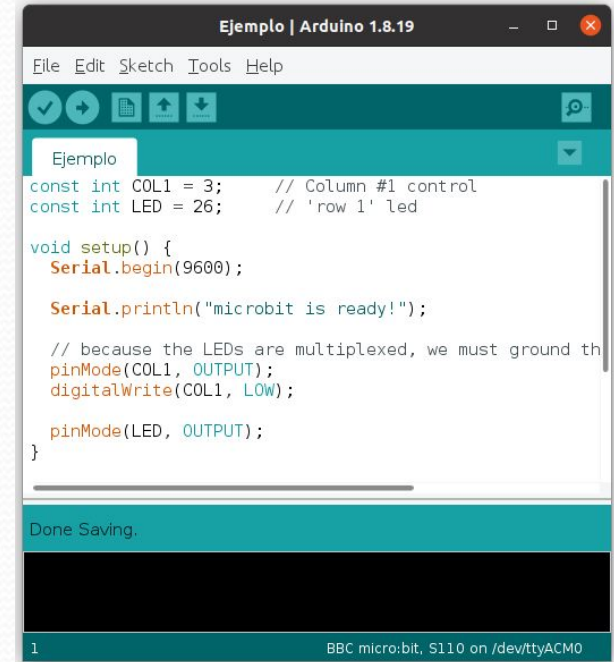
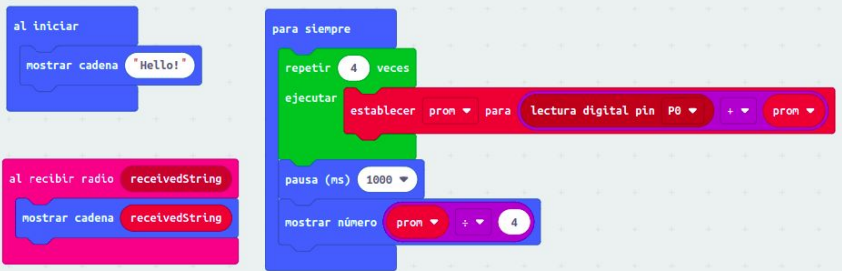
# Microcontroladores

- Alguna de sus muchas aplicaciones:
  - Biónica
  - Robótica
  - Domótica
  - Control vehicular
  - Control industrial (temperatura, ph, motores, etc)
  - Sintetizador de sonidos
  - Etc
  - Etc
  - Etc.....

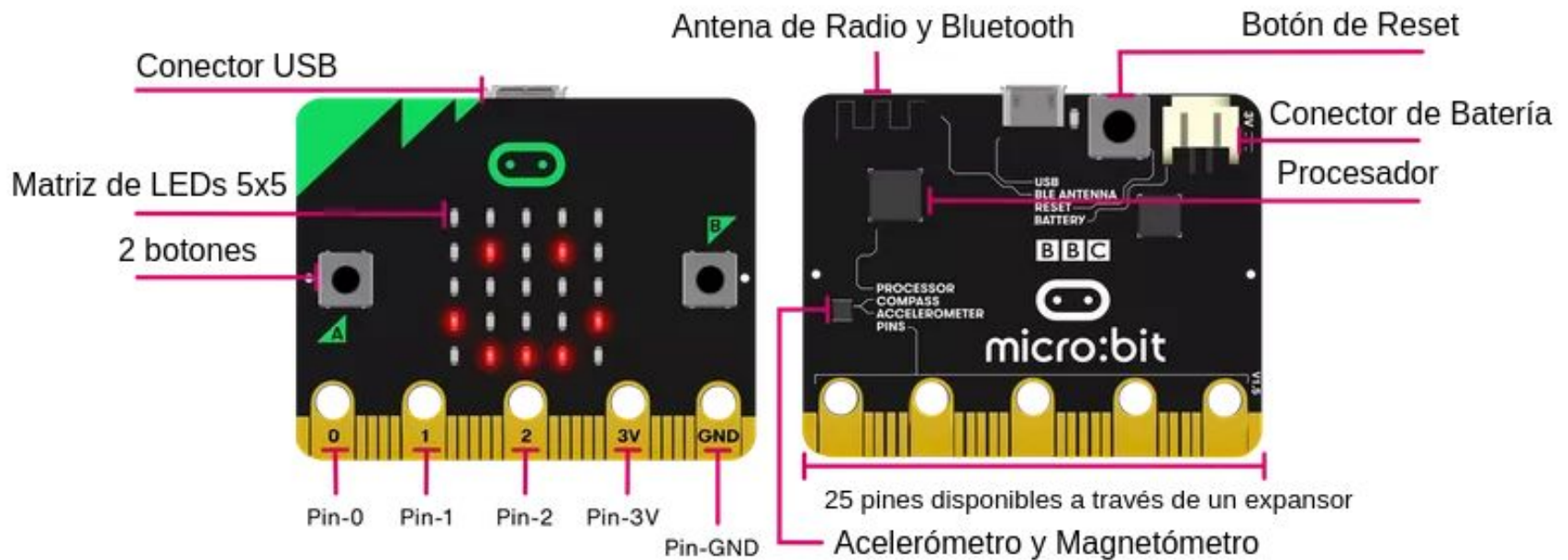
# Micro:bit

## ● ¿Cómo se programa?

- MakeCode
- Arduino IDE
- Micropython
- Otros



# Micro:bit v1

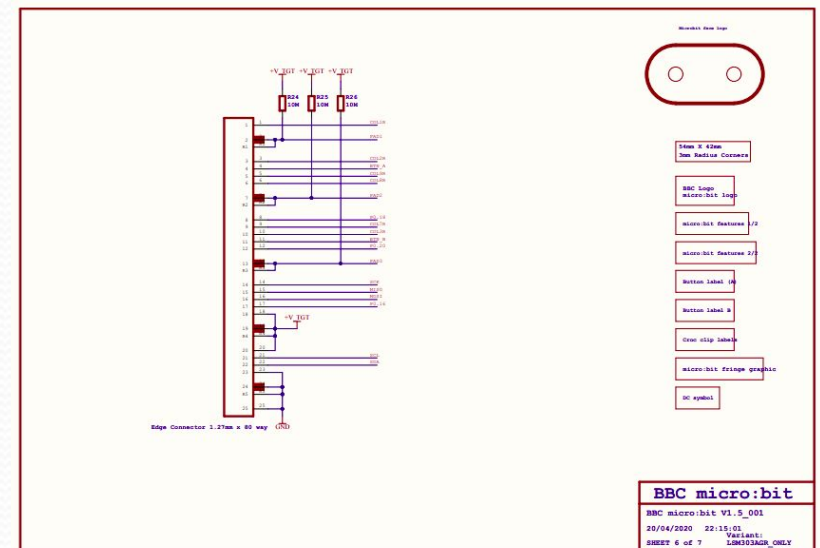
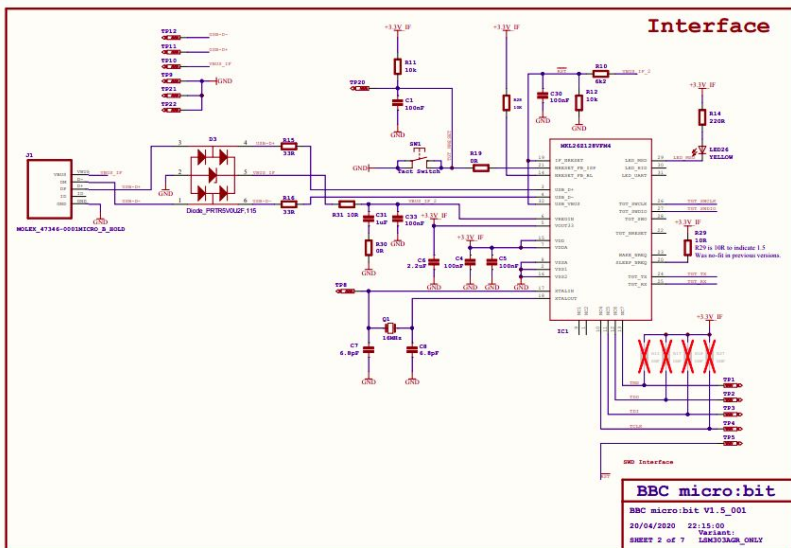
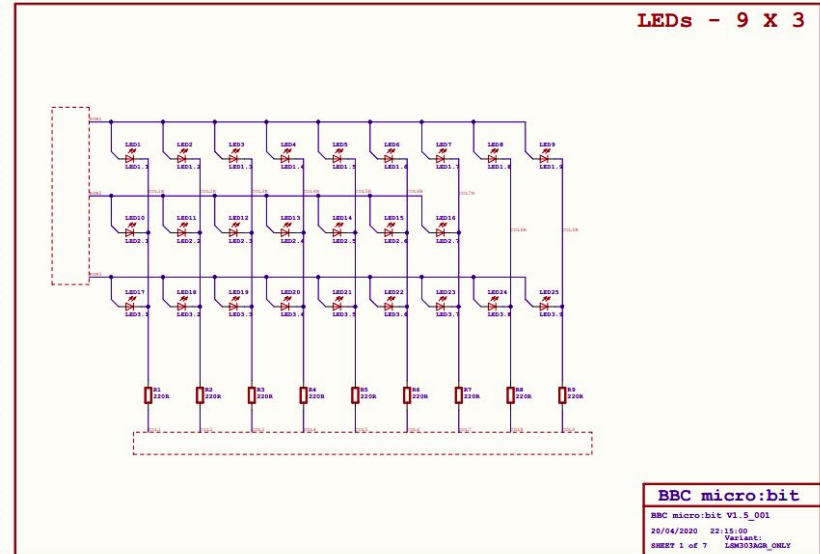
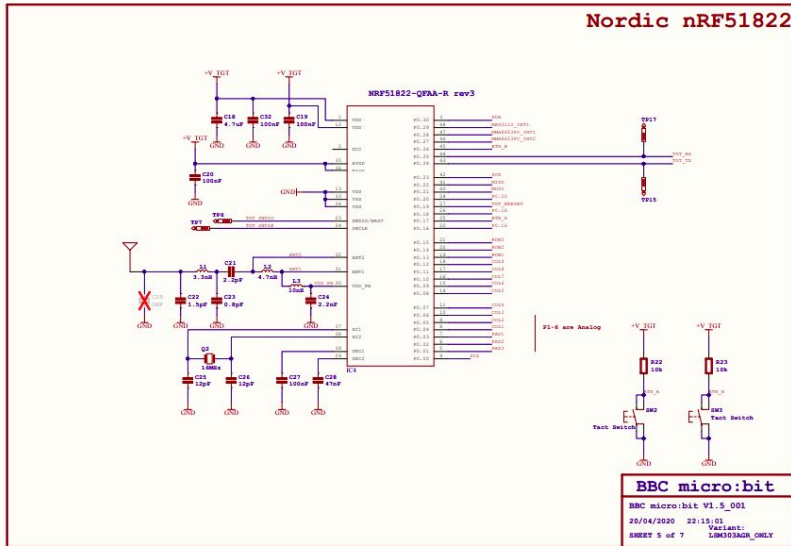


# Software / Hardware libre



- Sitio web: <http://www.arduino.cc>
- Filosofía Basada en **software/hardware** libre.
- Hardware libre:
  - Dispositivos de hardware cuyas especificaciones y diagramas esquemáticos son de acceso público.
  - <https://tech.microbit.org/hardware/>
- Software libre:
  - Libertad para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software.
  - <https://github.com/sandeepmistry/arduino-nRF5>

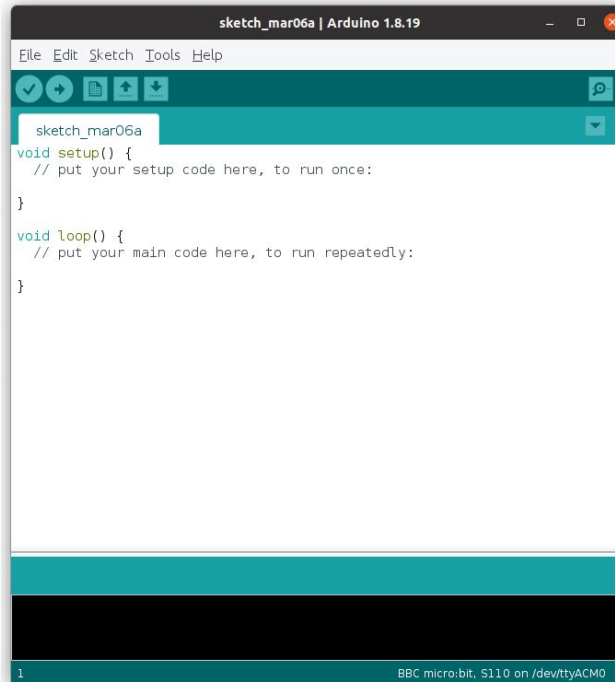
# Esquemático micro:bit v1



# Entorno de desarrollo: Arduino IDE

## ● Instalar entorno de desarrollo

- Ver página de curso:
  - *”Guía para configurar Arduino IDE para programar la placa micro:bit”*
- Versión actual: Arduino IDE 1.8.19.



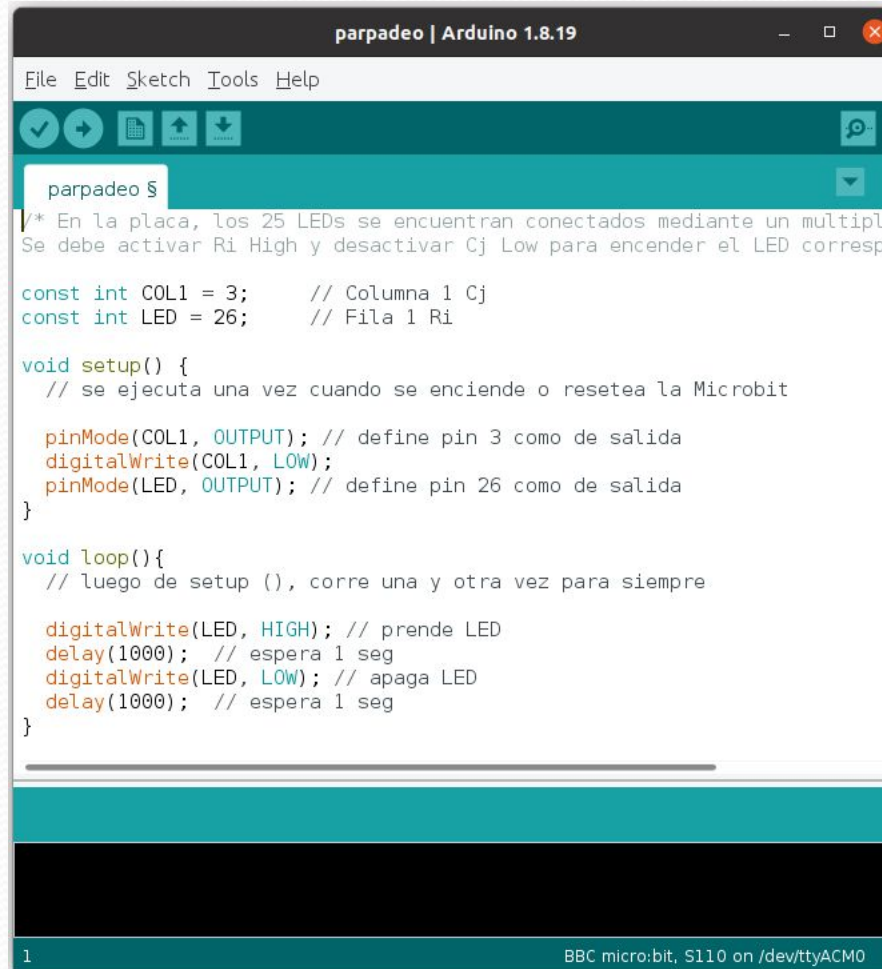


# Entorno de desarrollo: Arduino IDE

- Código Fuente (archivo “parpadeo.ino” suministrado)

Código fuente  
“sketch”

Lenguaje de alto  
nivel similar a C++



```
parpadeo 5
/* En la placa, los 25 LEDs se encuentran conectados mediante un multiplexor.
Se debe activar Ri High y desactivar Cj Low para encender el LED correspondiente. */

const int COL1 = 3;    // Columna 1 Cj
const int LED = 26;    // Fila 1 Ri

void setup() {
  // se ejecuta una vez cuando se enciende o resetea la Microbit

  pinMode(COL1, OUTPUT); // define pin 3 como de salida
  digitalWrite(COL1, LOW);
  pinMode(LED, OUTPUT); // define pin 26 como de salida
}

void loop(){
  // luego de setup (), corre una y otra vez para siempre

  digitalWrite(LED, HIGH); // prende LED
  delay(1000); // espera 1 seg
  digitalWrite(LED, LOW); // apaga LED
  delay(1000); // espera 1 seg
}

1 BBC micro:bit, S110 on /dev/ttyACM0
```

# Estructura de un sketch

```
int LED_LUZ = 13;
```

Declaración  
de variables

```
void setup () // esto es una FUNCION
{
    // esto se ejecuta una única vez
    pinMode (LED_LUZ , OUTPUT );
    // pin LED_LUZ salida de datos
}
```

Inicialización

```
void loop () // esto es otra FUNCION
{
    // todo esto se ejecuta una y otra vez
    digitalWrite (LED_LUZ ,LOW ); // apaga luz
    delay (1000); // espera 1000 milisegundos
    digitalWrite (LED_LUZ ,HIGH ); // prende luz
    delay (1000); // espera 1000 milisegundos
}
```

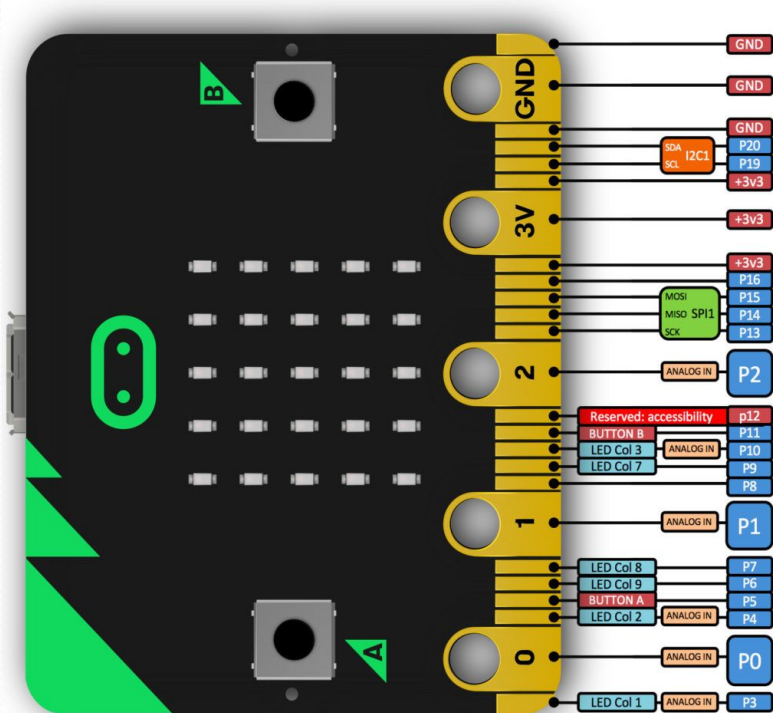
Ejecución del  
loop

# Entrada y Salida

- Comunicación de la micro:bit con el exterior
- Pines Digitales:
  - Pueden ser definidos como entrada o salida
    - Solo 2 valores posibles “1” o “0”
- Pines Analógicos:
  - Solo pueden definirse como entrada.
  - Conjunto continuo de valores.

Solo vamos a usar señales digitales

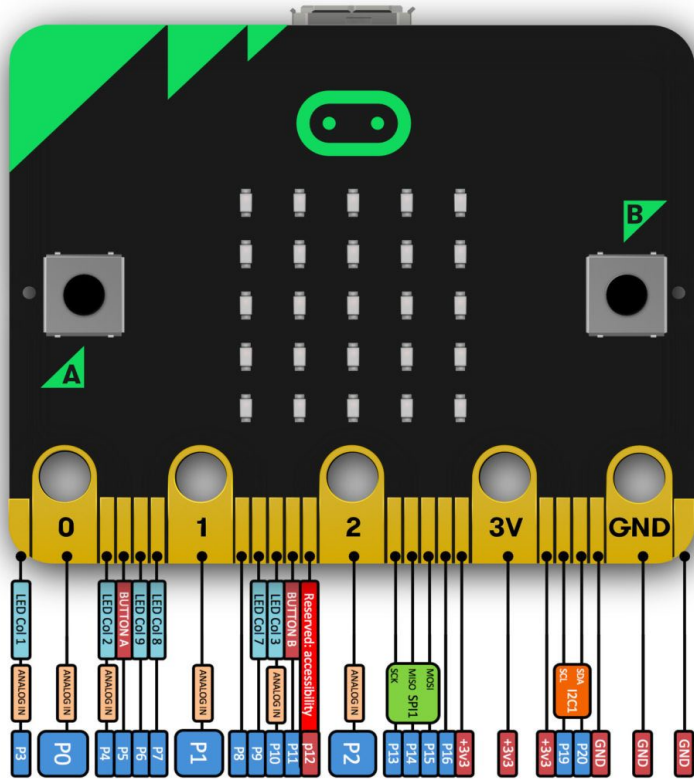
# Entrada y Salida



- P0, P1 y P2: Analógico / Digital
- P3, P4 y P10: Analógico / Digital / Matriz LEDs
- P5: Digital / Botón A
- P6 y P7 y P9: Digital / Matriz LEDs
- P8 y P16: Digital
- P12: reservada
- P11 Digital / Botón B
- P13, P14 y P15 Digital / SPI
- P19 y P20 Digital / I2C

Si se están usando los botones y la matriz de LEDs, quedan disponibles los pines: P0, P1, P2, P8, P12, P13, P14, P15, P16, P19 y P20.

# Entradas / Salidas Digitales



- 0 – 0,8V : es considerado como un “0” o FALSE
- 2 – 3,3V : es considerado como un “1” o TRUE
- 0,8V – 2V : ¿...?
  - Va a tomar un valor, “0” o “1” pero ..... no sabemos cuál !!!!

# Entradas / Salidas digitales

```
boolean llave = 0;
```

```
void setup (){  
  pinMode (10 , INPUT );  
  pinMode (13 , OUTPUT );  
}
```

```
void loop (){  
  llave = digitalRead (10); // lee el valor del pin 10  
                        // y lo guarda en "llave"  
  if( llave == LOW ){  
    digitalWrite ( 13 , HIGH ); // pone el pin 13 en +3V  
  }  
  else{  
    digitalWrite ( 13 , LOW ); // pone el pin 13 en +0V  
  }  
}
```

Pin 10 definido como entrada, "se lee"

Pin 13 definido como salida, "se escribe"

# Entradas / Salidas digitales

- Salida

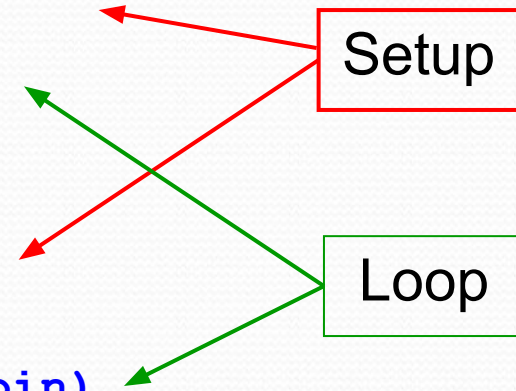
Declaración: `pinMode (pin, OUTPUT)`

Escritura: `digitalWrite (pin, valor)`

- Entrada

Declaración: `pinMode (pin, INPUT)`

Lectura: `Variable = digitalRead(pin)`



**pin:** es un número que identifica al número de PIN digital

**valor:** HIGH o LOW

**Variable:** donde se guarda el valor digital leído (se explicará más adelante)

# Entradas / Salidas digitales

¿Qué valor digital tendrá un pin de entrada si no lo conecto a nada?

R: **DESCONOCIDO**, podría ser “0” o podría ser “1”, pero.... no lo sabemos. Suele decirse que esta "flotando“.

El valor dependerá del nivel de voltaje presente en el pin, el cual puede variar al tocarlo con nuestros dedos o en función de interferencias electromagnéticas, etc.



# If-else if-else

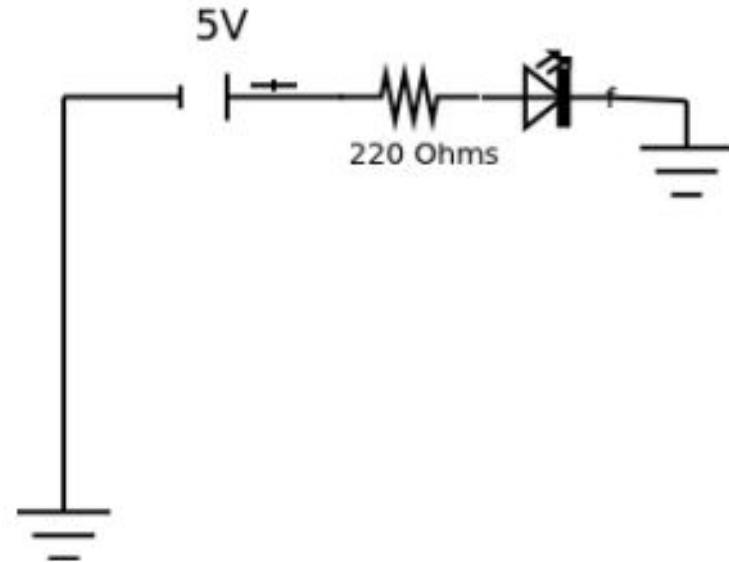
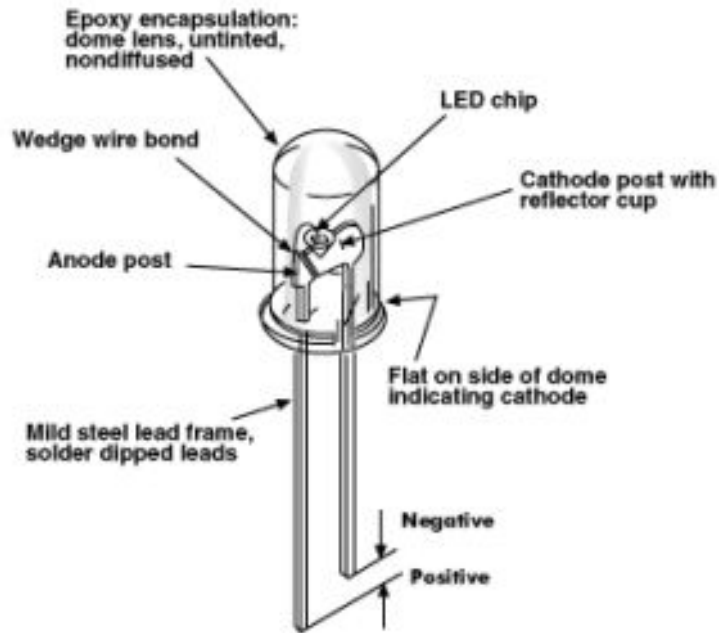
## if- else if-else

```
if (condición1=verdadera) {  
    Código1;  
} else if (condicion2=verdadera) {  
    Código2;  
} else {  
    Código 3;  
}
```

## EJEMPLO:

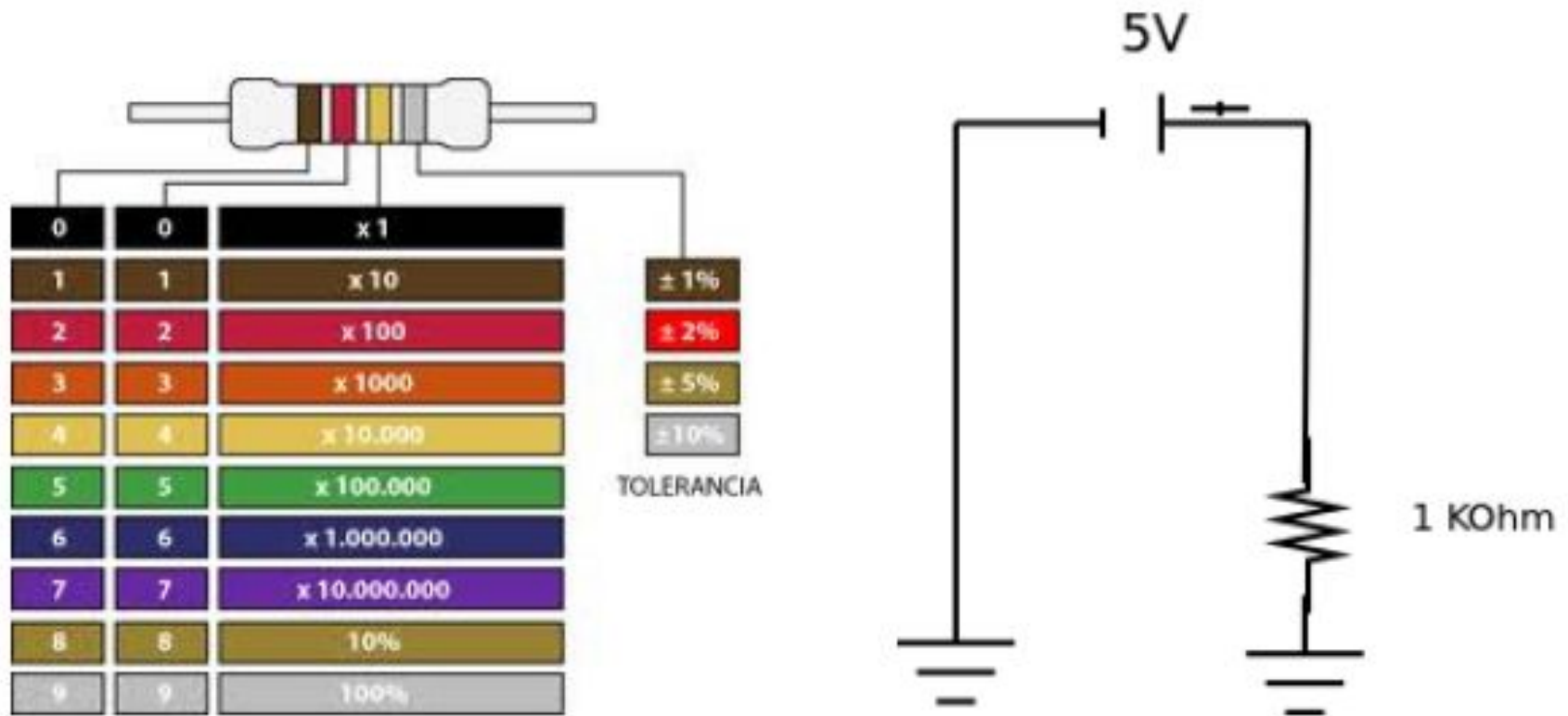
```
boolean llave = 0;  
  
void setup () {  
    pinMode (10 , INPUT );  
    pinMode (13 , OUTPUT );  
}  
  
void loop () {  
    llave = digitalRead (10);  
  
    if( llave == LOW ) {  
        digitalWrite ( 13 , HIGH );  
    }  
    else {  
        digitalWrite ( 13 , LOW );  
    }  
}
```

# LED - Light Emitting Diode



Descripción LED: [electroschematics.com](http://electroschematics.com)

# Resistencias



Código colores: [tecnorik.blogspot.com](http://tecnorik.blogspot.com)

# Comenzamos a trabajar!!!

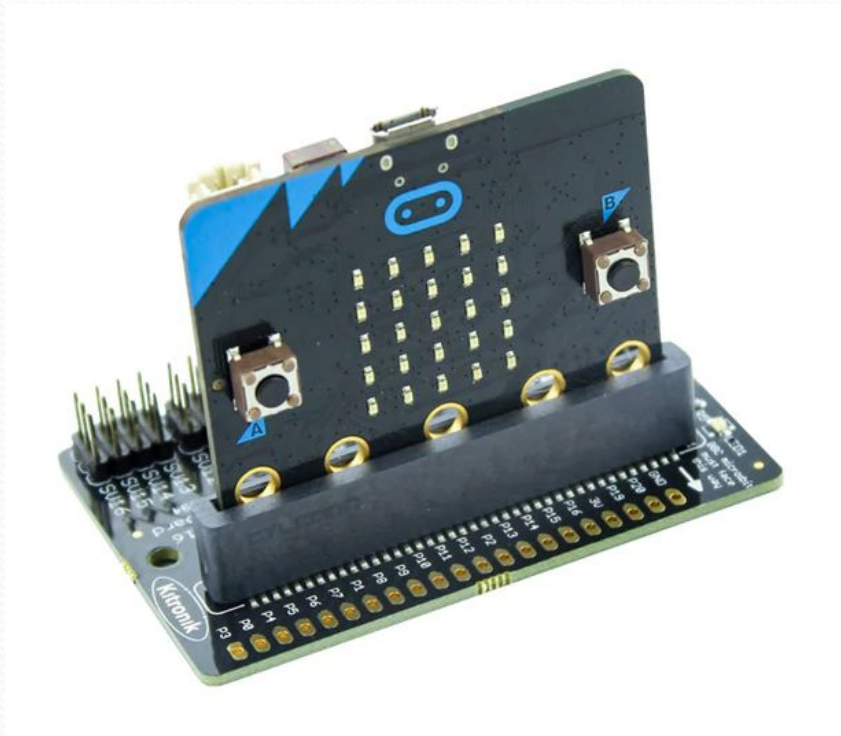
- Bajar de EVA/Tallerine Biónico/Semana 1:
  - “Guia para configurar Arduino IDE para programar la placa micro:bit”
  - Obligatorio 1
  - parpadeo.ino
  - parpadeo\_v2.ino
- Instalar el entorno de desarrollo. Seguir los pasos de la guía.
- Verificar que el material esté completo.
- Verificar **con un docente** el correcto funcionamiento del material.
- Comenzar con el Obligatorio 1.



# Comenzamos a trabajar!!!

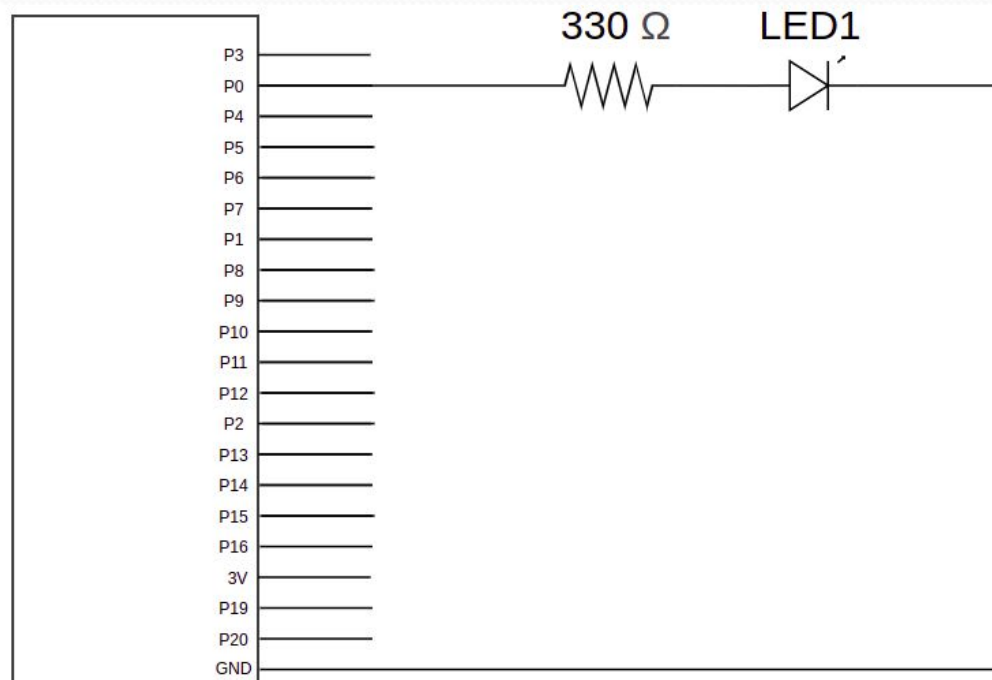
# Ejercicio 1

- Conectar la placa Kitronik como se observa en la figura. La micro:bit mirando hacia los pines P3, P0, .....



# Ejercicio 1 continuación

- Programar, compilar y cargar el programa `parpadeo_v2.ino`
- Implementar el circuito de la figura y observar lo que sucede al ejecutar el programa.
- Modificarlo para que el parpadeo sea cada 0.4 segundos



# Ejercicio 1 continuación

Verificar que la resistencia sea de 330 ohm !!

