

**ESPACIO CURRICULAR:** ELECTROTECNIA - ELECTROTECNIA Y ELECTRONICA

**DOCENTE 4to AyB:** FRECCERO, DANIEL GUSTAVO

**DOCENTE 4to C:** FRANCO, GONZALO

**TEMAS:** Arduino, entradas salidas, programacion

**OBJETIVOS:** Resolver situaciones problemáticas con los aprendizajes adquiridos, aprender no solo la teoría, sino demostrarlo de manera practica en clases, por eso es muy importante tu asistencia a las mismas.

**CRITERIOS DE EVALUACIÓN:**

- Tu correcta participación en Clases presenciales
- Demostrar de manera **práctica** los conocimientos teóricos
- Prolijidad en la entrega de las actividades, pasar las actividades a la carpeta y en las computadoras de la escuela



**FECHA DE ENTREGA:** 30/11/2022

## ¿Qué es Arduino?

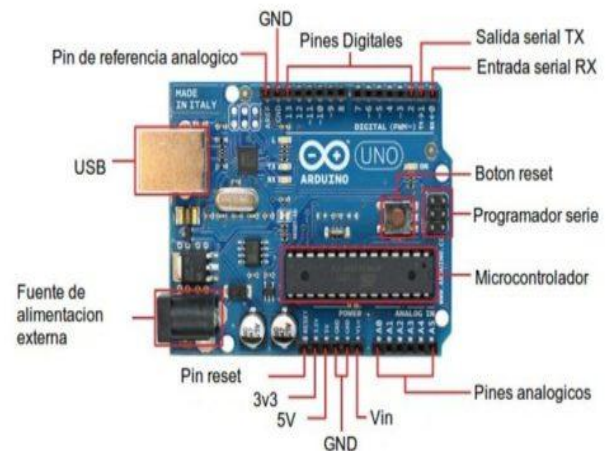
Arduino es una pequeña computadora que se puede programar para procesar entradas y salidas, hacia y desde el chip, es una plataforma de computación integrada, lo que significa que es un sistema interactivo, que a través del uso de hardware y el software pueden interactuar con él.

Un uso sencillo del Arduino sería enciende una luz por un período de tiempo establecido, digamos 30 Segundos, después de presionar un botón. En este ejemplo, el Arduino tendría una lámpara y un botón conectados; el Arduino se sentaría pacientemente a esperar a que se pulse el botón. Cuando se presione el botón se encendería la lámpara y comenzaría a contar 30 segundos después de contar 30 segundos apaga la lámpara y luego sigue sentado allí esperando a que se vuelva a presionar el botón.

## Estructura de código en Arduino

Un programa de Arduino se pueden dividir en tres partes principales: Estructura, valores (variables y constantes), y funciones. La estructura básica de programación de Arduino es bastante simple y divide la ejecución en dos partes: setup y loop. **Void Setup()** constituye la preparación del programa: En la función Setup() se incluye la declaración de variables y se trata de la primera función que se ejecuta en el programa. Esta función se ejecuta una única vez y es empleada para configurar el pinMode (p. ej. si un determinado pin digital es de entrada o salida)

**Void loop()** es la ejecución: La función loop() incluye el código a ser ejecutado continuamente (leyendo las entradas de la placa, salidas, etc.)



## Tipos de datos

Los tipos de datos en “lenguaje C” se refiere a un sistema amplio usado para declarar variables o funciones de diferentes tipos. **El tipo de una variable determina la cantidad de espacio que ocupa en el almacenamiento y cómo se interpreta el patrón de bits almacenado.**

Tipos de Datos	Memoria que ocupa	Rango de valores
boolean	1 byte	0 o 1 (True o False)
byte / unsigned char	1 byte	0 – 255
char	1 byte	-128 – 127
int	2 bytes	-32.768 – 32.767
word / unsigned int	2 bytes	0 – 65.535
long	2 bytes	-2.147.483.648 – 2.147.483.647
unsigned long	4 bytes	0 – 4.294.967.295
float / double	4 bytes	-3,4028235E+38 - 3,4028235E+38
string	1 byte + x	Array de caracteres
array	1 byte + x	Colección de variables

*Todo dato lo tengo que archivar en una variable, pero dependiendo de que sea, sera el espacio que le asigne, por ejemplo,, si pregunto, si es un puso alto o bajo, me sirve un tipo de dato booleano (0 o 1). Pero si pregunto la edad, tendria que necesitar un tipo de dato byte (0 a 255) si tendria que guardar el año que nacio, ya no me sirve, tendia que pasar a tipo int (hasta 32767). Ahorrando espacio puedo hacer mas grande mi programa. Para que lo entiendas de manera sencilla ¿Dónde guardarías una moneda? En un lugar sencillo, un monedero... pero no en un Armario... je*

## FUNCIONES

### **Funciones I/O digitales**

- `pinMode(pin, mode)`
- `digitalWrite(pin, value)`
- `int digitalRead(pin)`

### **Funciones I/O analógicas**

- `int analogRead(pin)`
- `analogWrite(pin, value)`

### **Funciones I/O avanzadas**

- `shiftOut(dataPin, clockPin, bitOrder, value)`
- `unsigned long pulseIn(pin, value)`

### **Funciones de tiempo**

- `unsigned long millis()`
- `delay(milisegundos)`
- `delayMicroseconds(microsegundos)`

### **Funciones matemáticas**

- `min(x, y)`
- `max(x, y)`
- `abs(x)`

- `constrain(x, a, b)`
- `map(value, fromLow, fromHigh, toLow, toHigh)`
- `pow(base, exponente)`
- `sq(x)`
- `sqrt(x)`
- `sin(rad)`
- `cos(rad)`
- `tan(rad)`

### **Funciones números aleatorios**

- `randomSeed(semilla)`
- `long random(max)`
- `long random(min, max)`

### **Comunicaciones serie**

- `Serial.begin(baudios)`
- `int Serial.available()`
- `int Serial.read()`
- `Serial.flush()`
- `Serial.print(datos)`
- `Serial.println(datos)`

# ESTRUCTURA

## Estructura principal

- void **setup()** (estructura de configuración)
- void **loop()** (estructura del bucle principal)

## Estructuras de control

- **if()**
- **if()...else**
- **for()**
- **switch()...case**
- **while()**
- **do()... while**
- **break**
- **continue**
- **return**
- **goto**

## Sintaxis

- **;** (punto y coma)
- **{}** (corchetes)
- **//** (comentario línea única)
- **/\* \*/** (comentario multilinea)

## Operadores aritméticos

- **=** (asignación)
- **+** (suma)
- **-** (resta)
- **\*** (multiplicación)
- **/** (división)
- **%** (módulo)

## Operadores de comparación

- **==** (igual que)
- **!=** (no igual que)
- **<** (menor que)
- **>** (mayor que)

- **<=** (menor o igual que)
- **>=** (mayor o igual que)

## Operadores booleanos

- **&&** (and)
- **||** (or)
- **!** (not)

## Operadores compuestos

- **++** (incremento)
- **--** (decremento)
- **+=** (suma compuesta)
- **-=** (resta compuesta)
- **\*=** (multiplicación compuesta)
- **/=** (división compuesta)

## Constantes

- **HIGH | LOW**
- **INPUT | OUTPUT**
- **true | false**

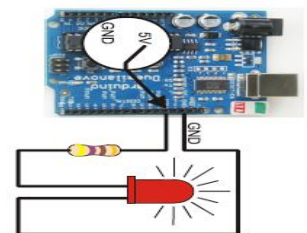
## Tipo de datos

- **boolean** (booleano)
- **char** (carácter)
- **byte** (byte)
- **int** (entero)
- **unsigned int** (entero sin signo)
- **long** (entero largo)
- **unsigned long** (entero largo sin signo)
- **float** (coma flotante)
- **double** (coma flotante doble)
- **string** (cadena de texto)
- **array** (array)
- **void** (nada)

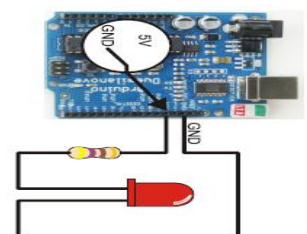
Veamos un ejemplo con las // puedo poner un comentario...

```
void setup() { //void setup es para definir cosas importantes
  pinMode(13, OUTPUT); // aquí digo que el pin 13 es de Salida
}

void loop() { // Menu principal, se autorrepite lo que tengo aquí
  digitalWrite(13, HIGH); // Pin 13 = 5 V, LED el led se enciende
  delay(500); // ..tiempo 0.5 segundos
  digitalWrite(13, LOW); // Pin 13 = 0 V, LED se apaga
  delay(500); // .. tiempo 0.5 segundos
}
```



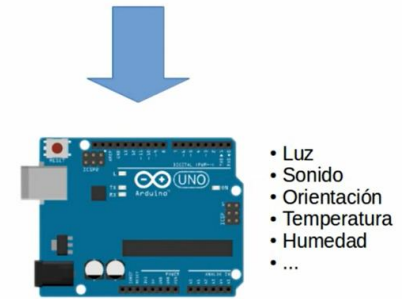
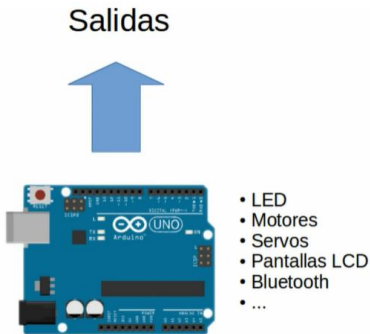
digitalWrite(13, HIGH);



digitalWrite(13, LOW);

¿Que son las entradas y salidas de Datos? Asi como tengo puertos para traer o enviar cosas a otros países, en el Arduino tengo puertos para entrar o salir datos.... Como ves en la imagen un dato de salida puede ser prender un led, emitir un sonido, hacer andar un motor... etc.

Información de los sensores



Y un dato de entrada puede ser un sensor, botón y muchas cosas más.

Los datos pueden ser digitales (0 que sería 0v y 1 que sería 5v) o Analógicos que varían entre 0 a 5 v, por ejemplo un medidor de humedad o temperatura.

```

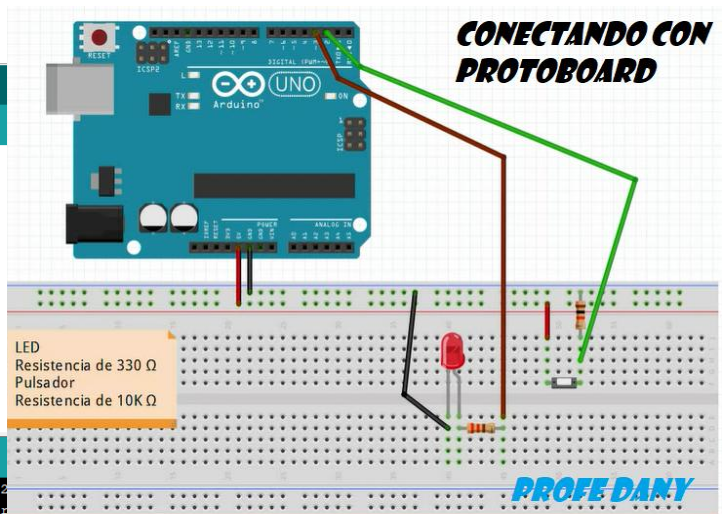
prendeapagaconboton Arduino 1.8.19
Archivo Editar Programa Herramientas Ayuda
prendeapagaconboton §
void setup() {
  //lo hace una sola vez
  pinMode(2, INPUT); // pin de ENTRADA
  pinMode(3, OUTPUT); // pin de SALIDA
}
void loop() {
  if (digitalRead(2)== HIGH) { //Pregundo si es la entrada un pulso alto y abro llave
    digitalWrite(3, HIGH); //si es asi... prende el led
  }
  else {
    digitalWrite(3, LOW); //sino... apaga el led
  }
}

```

Compilado

El Sketch usa 892 bytes (2%) del espacio de almacenamiento de programa. El máximo es 32...

Las variables Globales usan 9 bytes (0%) de la memoria dinámica, dejando 2039 bytes par...



Se realizaran 4 TPs con las Netbooks de la escuela y los Modulos Arduinos.

- 1) Prenden y apagar el led 13, integrado en el Arduino Uno
- 2) Hacer una secuencia de luces del Auto Fantastico con 5 led
- 3) Manejar entradas de Datos, mediante un swicht o botón
- 4) Hagamos un Semáforo doble con leds y si te es fácil adáptalo personas ciegas
- 5) Manejar Servomotores, aplicados a brazos roboticos

Para mas info... anda al Blog del Profe Dany <https://roboticaconarduino132.blogspot.com/>

Si no te sale, en clases lo veremos, Saludos Profes Daniel Freccero y Gonzalo Franco

Saludos!