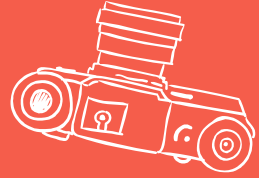


CLASE 1

ARQUI 1 (SECCIÓN B)





HOLA!

José Fernando Valdez Pérez
María de los Angeles Herrera Sumalé

Grupo de whatsapp:

<https://chat.whatsapp.com/J6CYUmZCgxiElDdSSTGAiv>

Código de clase de classroom: d4ibnjr

Pueden enviarnos sus dudas con el asunto [ACYE1]Asunto a:

valdezpjose@gmail.com

marielos.10h@gmail.com

PONDERACIÓN

3

ACTIVIDAD	PUNTEO
Practica 1 Arduino	5
Practica 2 Arduino	5
Proyecto Arduino	15
Practica 1 Assembler	10
Practica 2 Assembler	15
Proyecto Assembler	20
Tareas/Hojas de Trabajo	10
Exámenes Cortos	5
Exámen Final	10
TOTAL	100



CALENDARIZACIÓN

ACTIVIDAD	FECHA INICIO	FECHA FIN
Practica 1 Arduino	05/02	12/02
Practica 2 Arduino	12/02	19/02
Proyecto Arduino	19/02	12/03
Practica 1 Assembler	12/03	24/03
Practica 2 Assembler	25/03	09/04
Proyecto Assembler	09/04	06/05



MICROPROCESADORES

También conocido como CPU o unidad central de procesamiento



¿QUE ES UN MICROPROCESADOR?

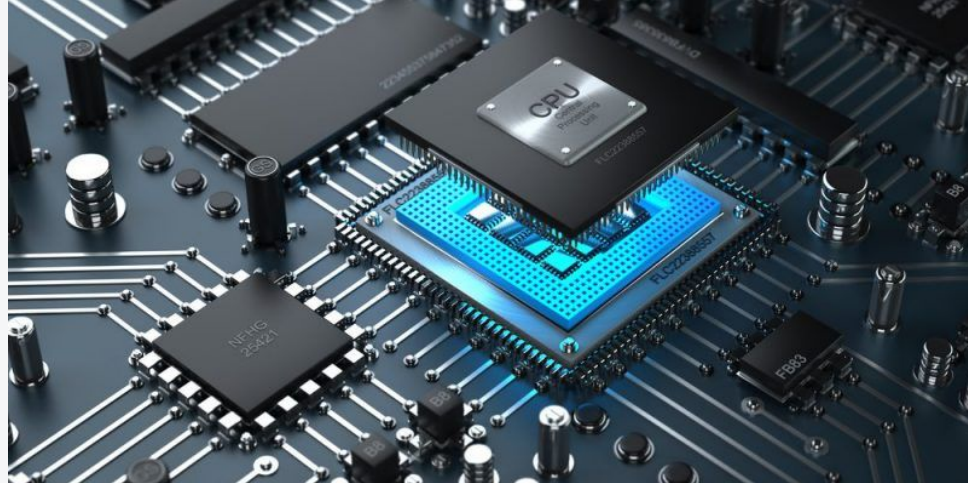
✗ Es el elemento controlador de un sistema computacional

✗ Es un chip o pastilla de silicio, que contiene circuitos integrados, transistores, y que se halla protegido por una cobertura de cerámica y plástico.

✗ Es un circuito lógico que responde y procesa las operaciones lógicas y aritméticas que hacen funcionar a nuestras computadoras



Un microprocesador está compuesto por varios bloques interconectados entre sí pero cada uno de ellos tiene una función específica que cumplir, al diseño e interconexión de estos bloques se le denomina Arquitectura



Es el encargado de ejecutar los programas, desde el sistema operativo hasta las aplicaciones de usuario; sólo ejecuta instrucciones programadas en lenguaje de bajo nivel, realizando operaciones aritméticas y lógicas simples, tales como sumar, restar, multiplicar, dividir, las lógicas binarias y accesos a memoria.



CARACTERISTICAS



Realiza todas las operaciones de cálculo



Controla lo que pasa en el ordenador dando órdenes a los demás componentes para que trabajen



Es el cerebro del ordenador



Tienen una alta velocidad de procesamiento



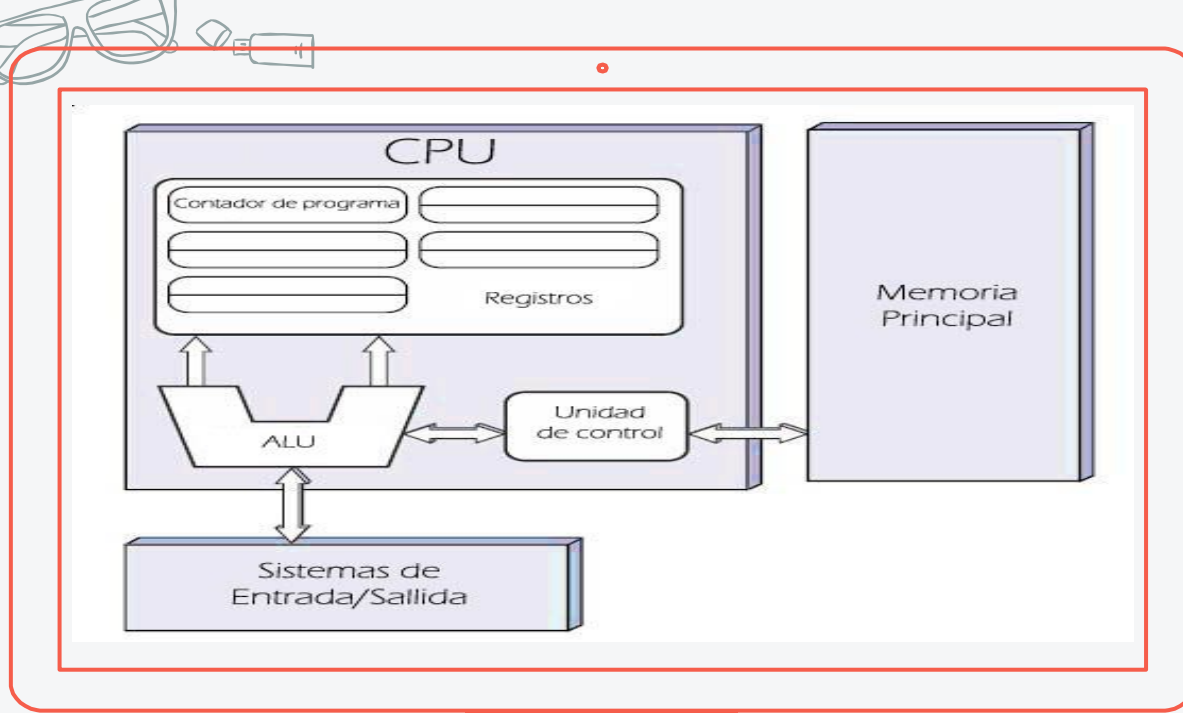
Son Eficientes en el microcódigo



Capaz de ejecutar las órdenes grabadas en su memoria

ARQUITECTURAS PARA COMPUTADORES





ARQUITECTURA VON NEUNMAN

Hoy en dia la mayoría de las computadoras son construidas con esta arquitectutra

ARQUITECTURA VON NEUNMAN - PARTES FUNDAMENTALES

✗ Unidad de Memoria

✗ Unidad de entrada y salida

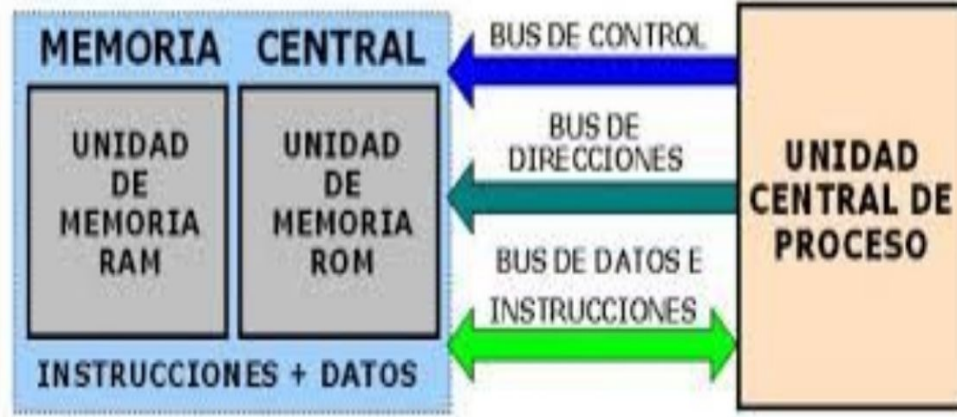
✗ Unidad central de proceso (CPU)

✗ Unidad de control (CU)

✗ Unidad Aritmetico Logica (ALU)

✗ Registros

ARQUITECTURA VON NEUMANN

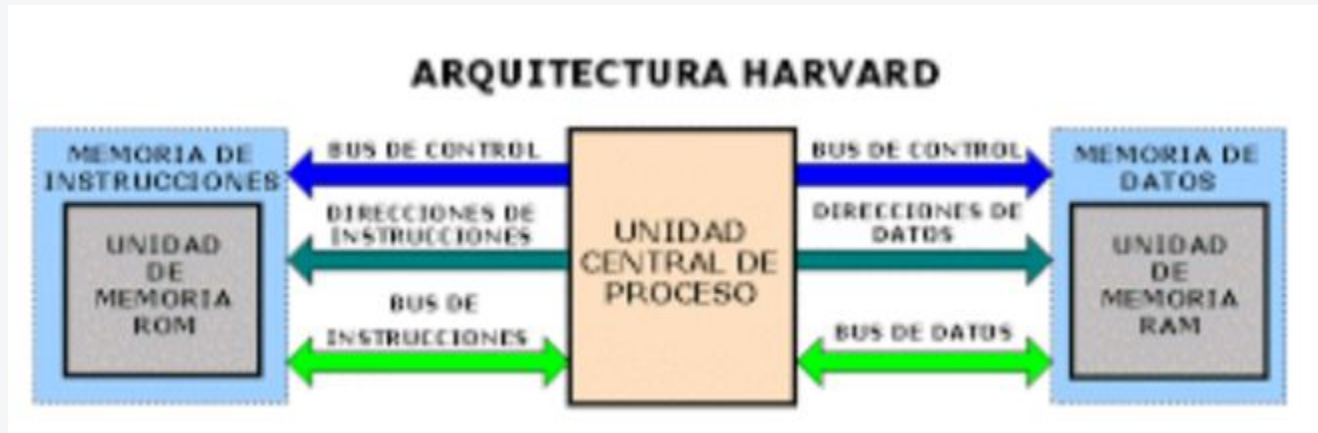


ARQUITECTURA VON NEUMANN

=

ARQUITECTURA HARVARD

En esta arquitectura se utilizan dispositivos de almacenamiento (memorias) separados para las instrucciones y los datos, y tiene dos sistemas completos de buses, uno para datos y otro para instrucciones. Esta arquitectura permite llevar simultáneamente datos e instrucciones por lo que permite mayor rapidez.



VENTAJAS

VON NEUMANN

1. La unidad de control recupera los datos y las instrucciones de la misma manera desde la memoria. Por tanto, el diseño y desarrollo de la unidad de control está simplificado, siendo más barato y más rápido.
2. Los datos de los dispositivos de entrada/salida y de la memoria principal se recuperan de la misma manera.
3. El diseño del chip del microcontrolador es mucho más sencillo, ya que se accederá a una sola memoria. Lo más importante del microcontrolador es el acceso a la RAM y en la arquitectura von Neumann esta se podrá usar tanto para almacenar datos como para almacenar instrucciones del programa.

HARVARD

1. Se tiene un mayor ancho de banda de memoria, que es más predecible por tener memorias separadas para las instrucciones y datos.
2. Las dos memorias pueden usar diferentes tamaños de celda, con lo cual se hace un uso efectivo de los recursos.
3. Hay menos posibilidades de corrupción en la transmisión, ya que los datos y las instrucciones se transfieren a través de diferentes buses.

DESVENTAJAS

VON NEUMANN

1. Por el procesamiento secuencial de las instrucciones no se permite la implementación paralela del programa.
2. Al compartir la memoria existe el riesgo que se escriba una instrucción sobre otra debido a un error en el programa, haciendo que se bloquee el sistema.
3. Algunos programas con defectos no pueden liberar la memoria cuando terminan con ella, lo que podría causar que se bloquee la computadora debido a que la memoria se haga insuficiente.

HARVARD

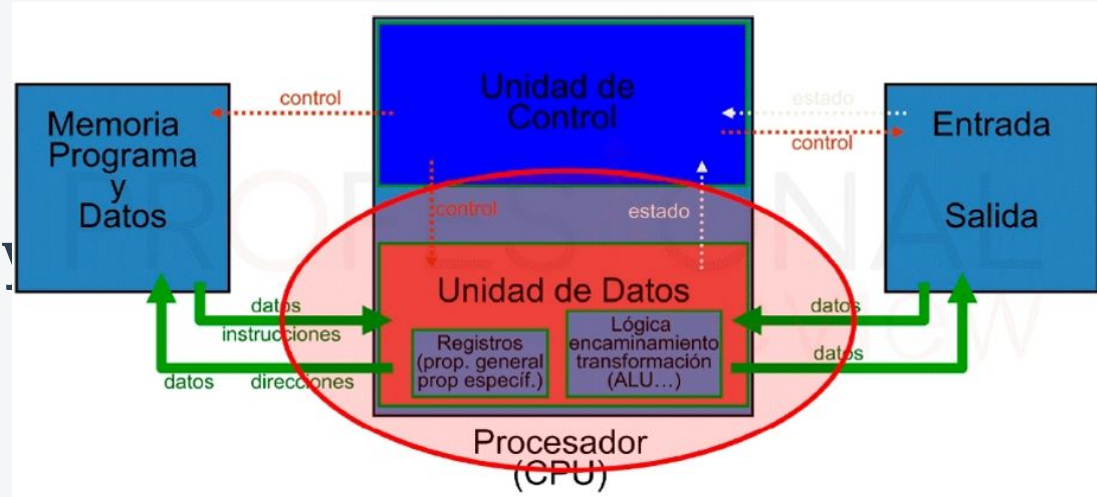
1. Mayor complejidad y costo: El problema con la arquitectura Harvard es su gran complejidad y costo debido a que, en lugar de un bus de datos, ahora se necesitan dos.
2. La producción de una computadora con dos buses es mucho más costosa y lleva más tiempo fabricarla. Requiere una unidad de control para dos buses, que es más complicada y cuyo desarrollo es costoso y necesita más tiempo.
3. Poca utilización: La arquitectura Harvard no se usa mucho, por lo que es más difícil de implementar. Es por eso que rara vez se usa fuera de la CPU. Se utiliza dentro de la CPU para manejar sus cachés

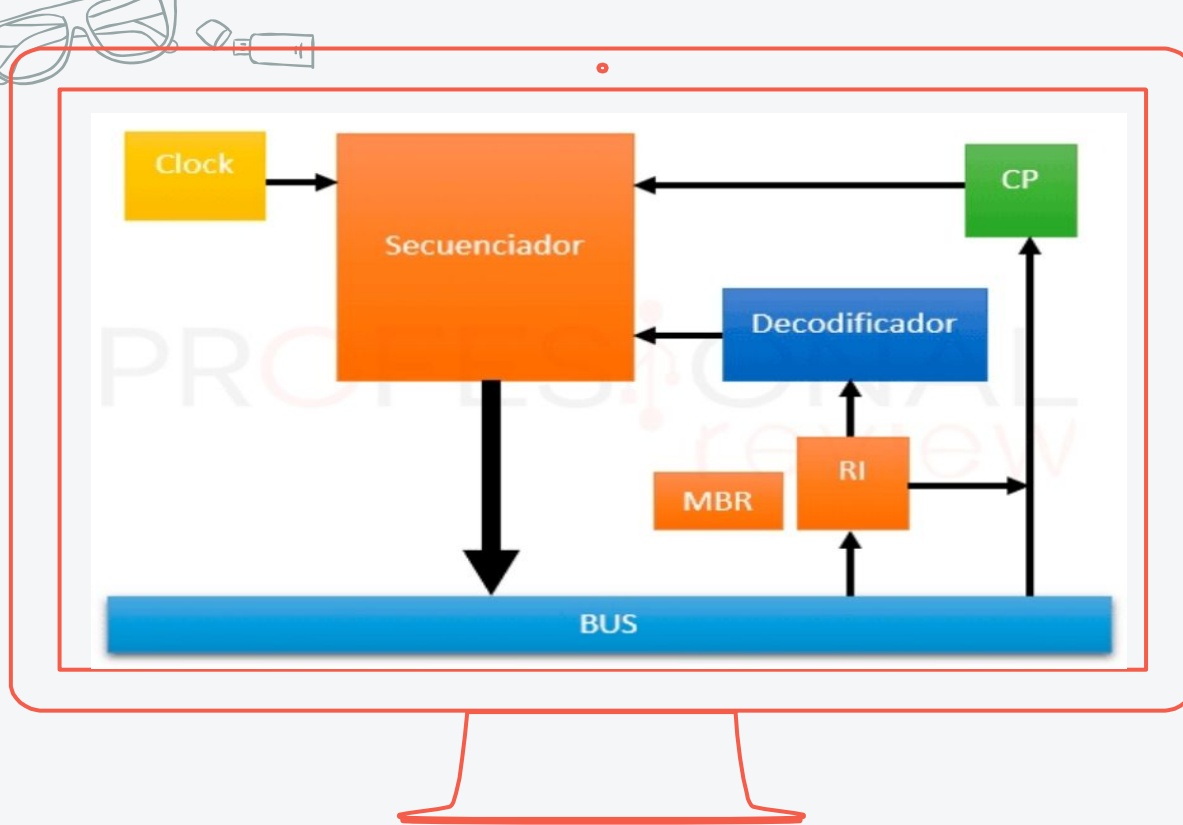


PARTES DE UNA COMPUTADORA



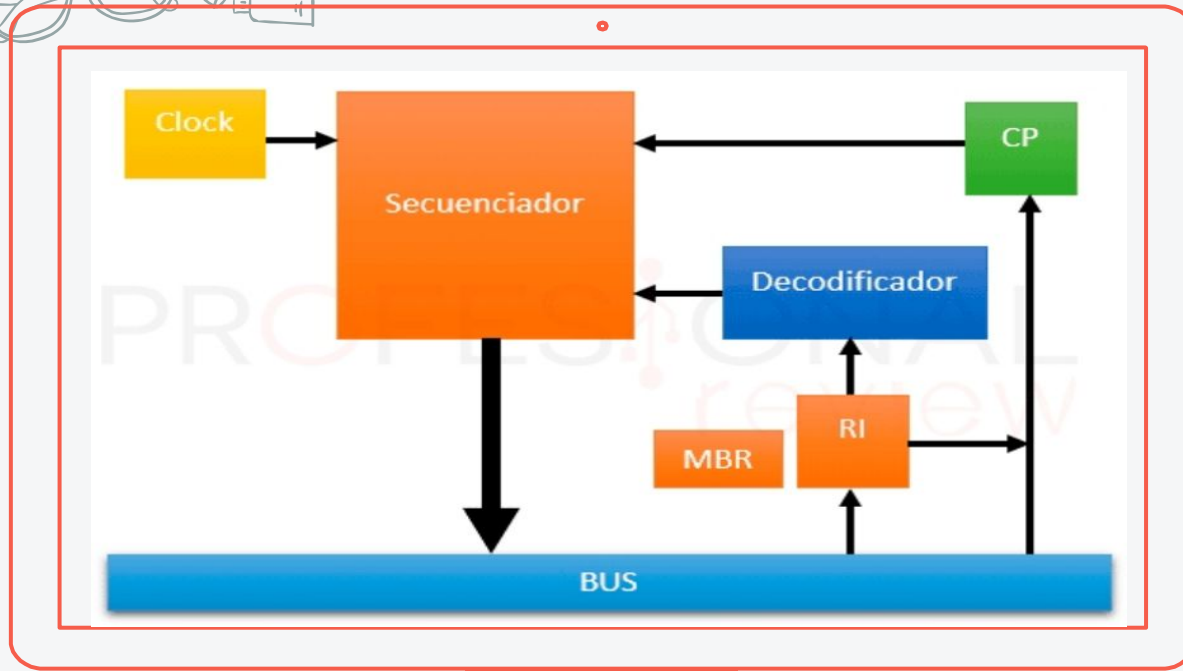
- ❌ Memoria
- ❌ Unidad Central de Procesamiento
- ❌ Unidad de entrada y salida
- ❌ Buses de datos





UNIDAD DE CONTROL UC

Es el elemento que se encarga de impartir las órdenes mediante las señales de control

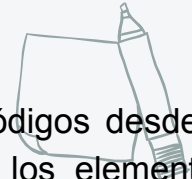


UNIDAD ARITMETICO LOGICA

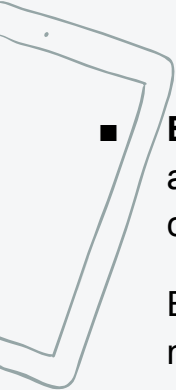
Esta parte de la arquitectura está involucrada únicamente en la realización de operaciones aritméticas y lógicas sobre los datos. Estarán disponibles los cálculos habituales como sumar, multiplicar, dividir y restar, pero también estarán disponibles las comparaciones de datos como 'mayor que', 'menor que', 'igual a'.



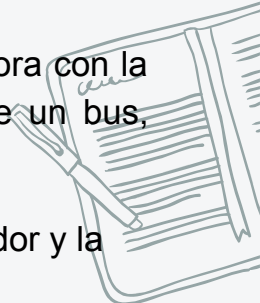
OTRAS PARTES



- **Sistema de entrada y salida:** Genera las señales necesarias para transferir datos y códigos desde y hacia periféricos. Un periférico es aquel dispositivo que es capaz de interactuar con los elementos externos ya sea emitiendo información o recibéndola.



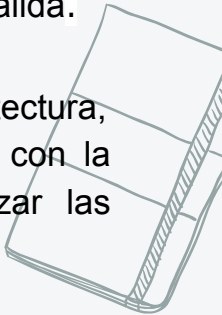
- **Bus:** La información debe fluir entre las diferentes partes de la computadora. En una computadora con la arquitectura von Neumann, la información se transmite de un dispositivo a otro a lo largo de un bus, conectando todas las unidades de la CPU a la memoria principal.



El bus de direcciones transporta las direcciones de los datos, pero no los datos, entre el procesador y la memoria.

El bus de datos transporta los datos entre el procesador, la memoria y los dispositivos de entrada-salida.

- **Unidad de coma flotante (FPU):** este elemento no estaba en el diseño original de la arquitectura, posteriormente fue introducido cuando las instrucciones y cálculos se hicieron más complejos con la aparición de los programas representados gráficamente. Esta unidad se encarga de realizar las operaciones en coma flotante, es decir, números reales.

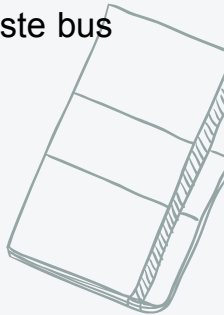


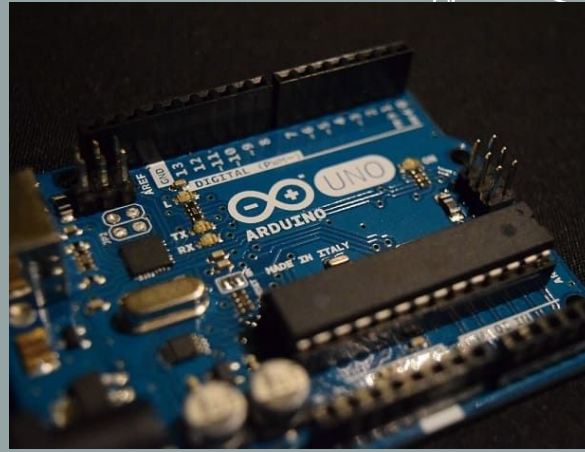


OTRAS PARTES



- **Banco de registros y la memoria caché (Caché):** los procesadores actuales cuentan con memoria volátil que hace de puente desde la memoria RAM hasta la CPU. Esta es mucho más rápida que la memoria RAM y se encarga de acelerar los accesos del microprocesador a la memoria principal.
- **Bus frontal (Front Side Bus, FSB):** También se conoce como bus de datos, bus principal o bus de sistema. Es la vía o canal que comunica el microprocesador con la placa base.
- **Bus trasero (Back Side Bus, BSB):** Este bus comunica la memoria caché de nivel 2(L2) con el procesador, siempre y cuando esta no esté integrada en el propio núcleo de la CPU. En la actualidad todos los microprocesadores disponen de memoria cache integrada en el propio chip, por lo que este bus también forma parte del mismo chip.






MICROCONTROLADORES

Un microcontrolador al menos tendrá:


1. **Microprocesador.**
2. **Periféricos (unidades de entrada/salida).**
3. **Memoria.**



Los microcontroladores están “ocultos” dentro de un sorprendente número de dispositivos electrónicos, desde microondas hasta televisores, drones y teléfonos inteligentes. En esta nueva era de la Industria 4.0, popularmente conocida como Internet de las Cosas e Internet of Things (IoT), los microcontroladores son una interesante solución en el campo de la electrónica con una arquitectura eficiente para soportar una amplia gama de opciones de conectividad.

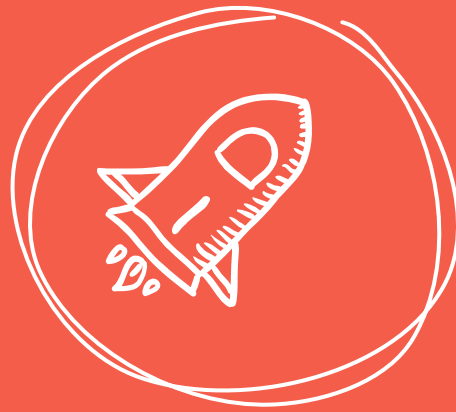


Un microcontrolador (MCU) utiliza técnicas de microelectrónica para reducir varios componentes como la CPU (Unidad de Procesamiento Central) y la memoria a un pequeño paquete (o package). Tiene varios pines de entrada y salida, a través de los cuales se puede interactuar con el mundo exterior. Obviamente, el microcontrolador como se ha realizado no hace nada, es necesario programarlo mediante un conjunto de instrucciones que representan el firmware del sistema instalado en la memoria del dispositivo.



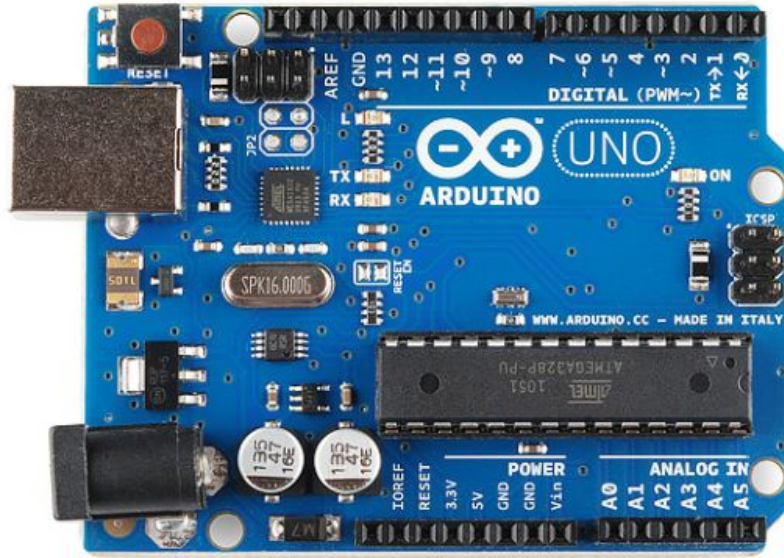
Puedes pensar en un microcontrolador como un pequeño ordenador; puedes conectar una pantalla, algunos botones, un motor y algunos sensores como acelerómetros y giroscopios, todos programados para realizar ciertas funciones.





ARDUINO





```

Blink
Blink
Turns on an LED on for one second, then off for one second, repeatedly.

This example code is in the public domain.
*/


// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(3000);             // wait for a second
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW
  delay(3000);             // wait for a second
}

```

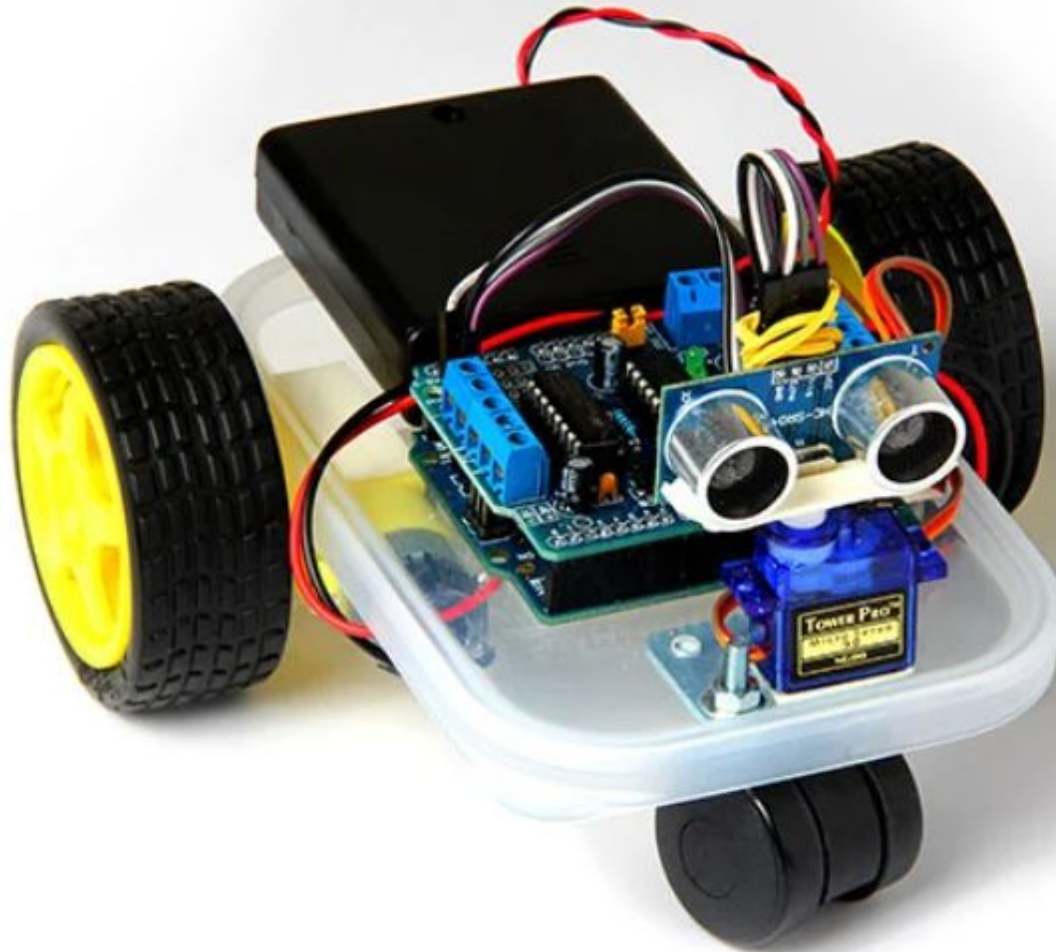
Arduino Mega (ATmega2560 on /dev/tty.usbserial-A800wrtd)



Es una plataforma de hardware libre, basada en una placa con un microcontrolador y un entorno de desarrollo (IDE, por sus siglas en Ingles), diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios.


Un Arduino es una placa que cuenta con todos los elementos necesarios para conectar periféricos a las entradas y salidas del microcontrolador






¿Qué hace?

El hardware y el software de Arduino fue diseñado para artistas, diseñadores, aficionados, hackers novatos, y cualquier persona interesada en la creación de proyectos electrónicos. Arduino puede interactuar con botones, LEDs, motores, altavoces, unidades de GPS, cámaras, internet, e incluso smartphones.






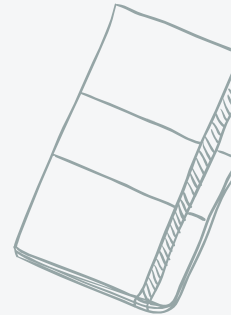
Es una placa de entrada / salida física (E / S) con un circuito integrado programable (IC). También incluye el entorno de desarrollo integrado (IDE) para la programación.



Funciones principales:

- ✗ I/O Digitales
- ✗ I/O Análogas
- ✗ Tiempo

Algunas constantes:

- ✗ High
 - ✗ Low
 - ✗ True
 - ✗ False
 - ✗ Input
 - ✗ OutPut
 - ✗ Constantes numéricas
- 
- 
- 
- 

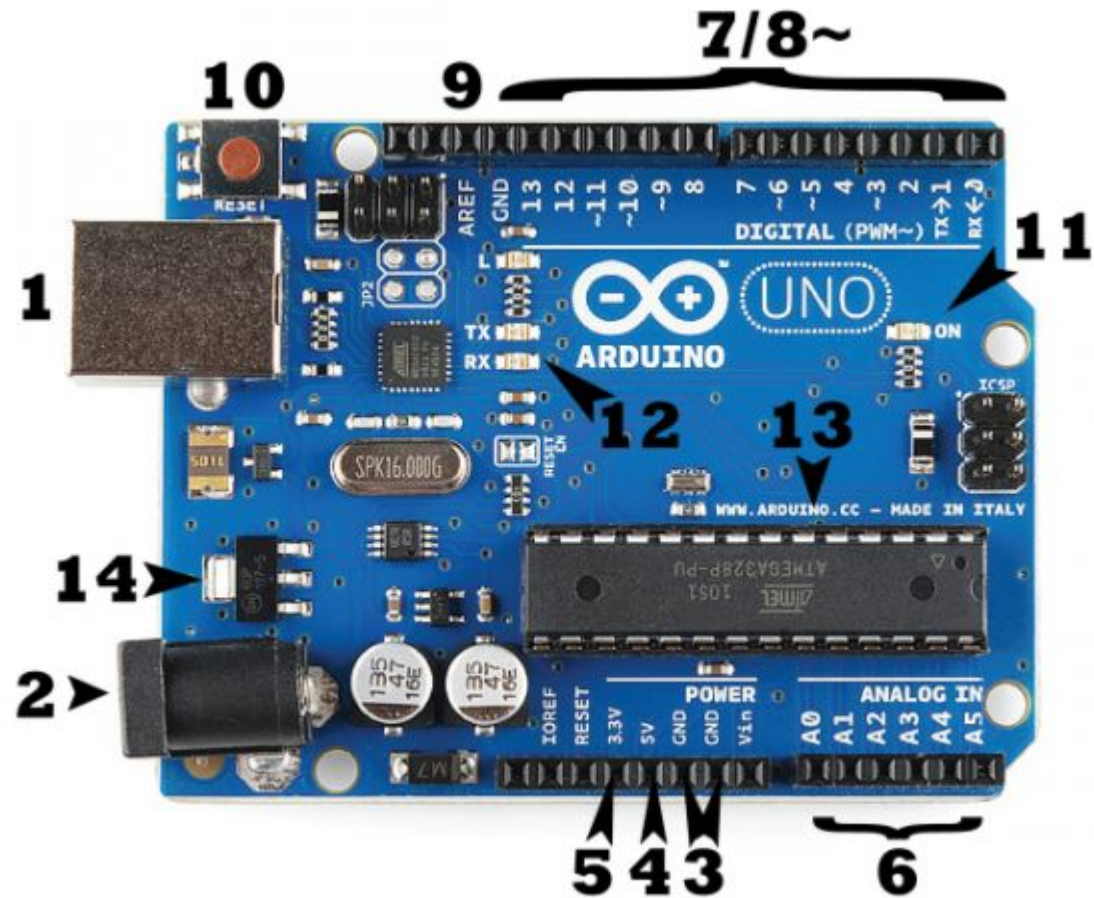


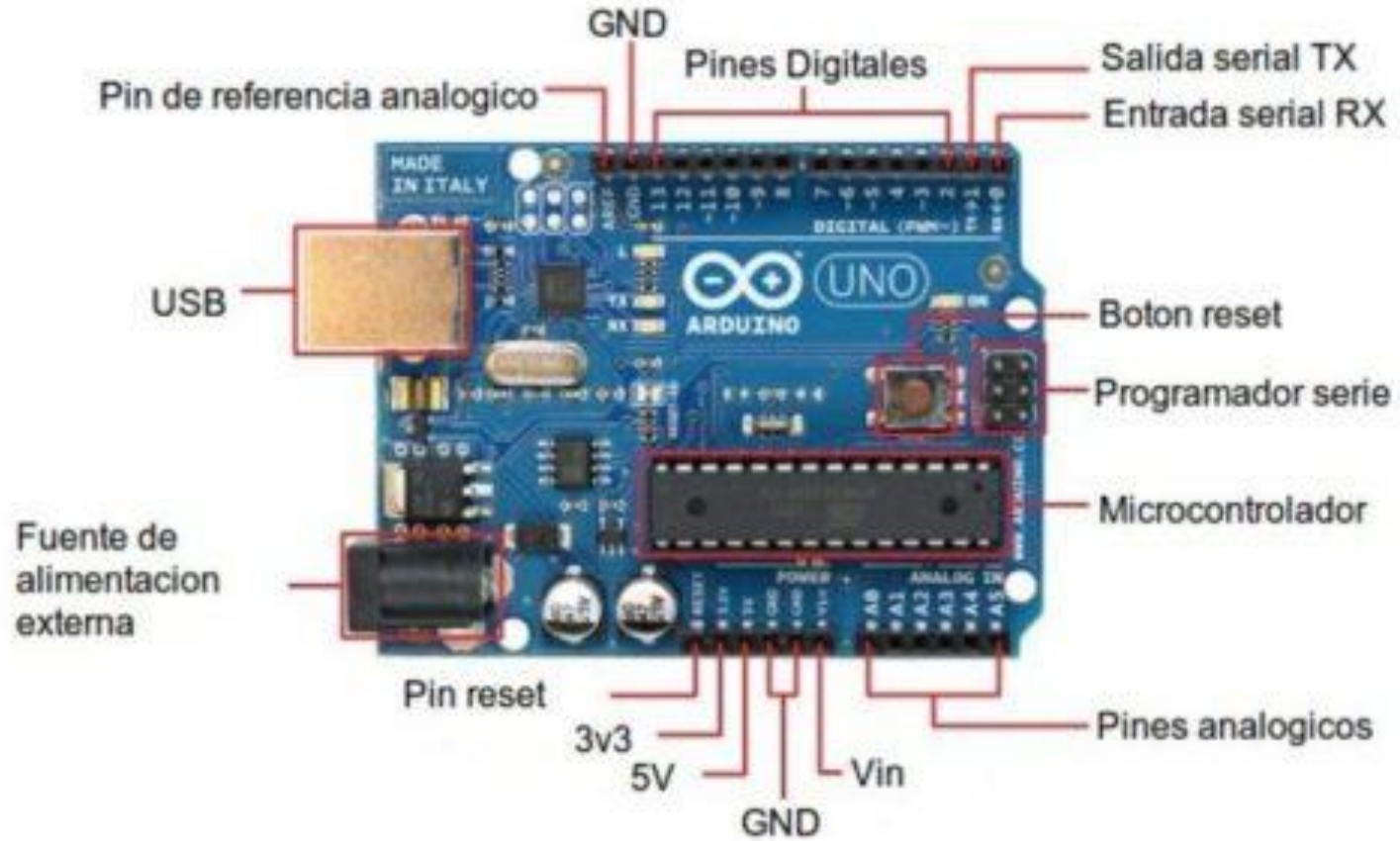
TIPOS DE ARDUINO



- ❖ Arduino Nano
- ❖ Arduino UNO
- ❖ Arduino DUE
- ❖ Arduino Leonardo
- ❖ Arduino Mega 2560
- ❖ Arduino Mega ADK
- ❖ Arduino Micro
- ❖ Arduino YUN
- ❖ Arduino FIO

PARTES DE UN ARDUINO





ARDUINO / PROTEUS

Arduino /
IDE

Proteus / Programa de
simulación



MATERIAL DE APOYO:

- ❖ Instalación Proteus: https://youtu.be/v_QXdQmXJCw
- ❖ Instalación librería Arduino en proteus: <https://youtu.be/pSu2TwwXFMk>
- ❖ Curso Arduino: <https://youtu.be/pSu2TwwXFMk>

Tarea 1:

La tarea consiste en instalar proteus y arduino/ide luego deberán enviar los siguientes entregables en un pdf:

1. Screenshoot de proteus instalado
2. Screenshoot de arduino/ide instalado
3. Programa de un led que se prende y apaga de forma intermitente (archivo de proteus)
4. Código fuente que usaron en arduino IDE

¡GRACIAS POR SU ATENCION!

¿Dudas?