Practica 2 Microcontroladores familia INTEL 8031-51-52

Diagrama de pines Intel 8031/8051

• El INTEL 805 F es un microcontrolador de 8 bits con 128 bytes de RAM interna y 4kb de ROM interna.

• El INTEL 8031 es el mismo que el 8051 excepto que no tiene ROM interna.

• El 8051 es un IC de 40 pines disponible en paquete de línea de doble entrada (DIP).

• Requiere una sola fuente de alimentación de +5V.

• Su frecuencia máxima de reloj interno es de 12 MHz

Mackenzie no es muy claro solo habla del 8051 y no refleja mucho acerca de los otros micros

[Pins\_and\_signals\_of\_8031\_8051\_microcontroller.pdf (idc-online.com)](https://www.idc-online.com/technical_references/pdfs/electronic_engineering/Pins_and_signals_of_8031_8051_microcontroller.pdf) <-fuente

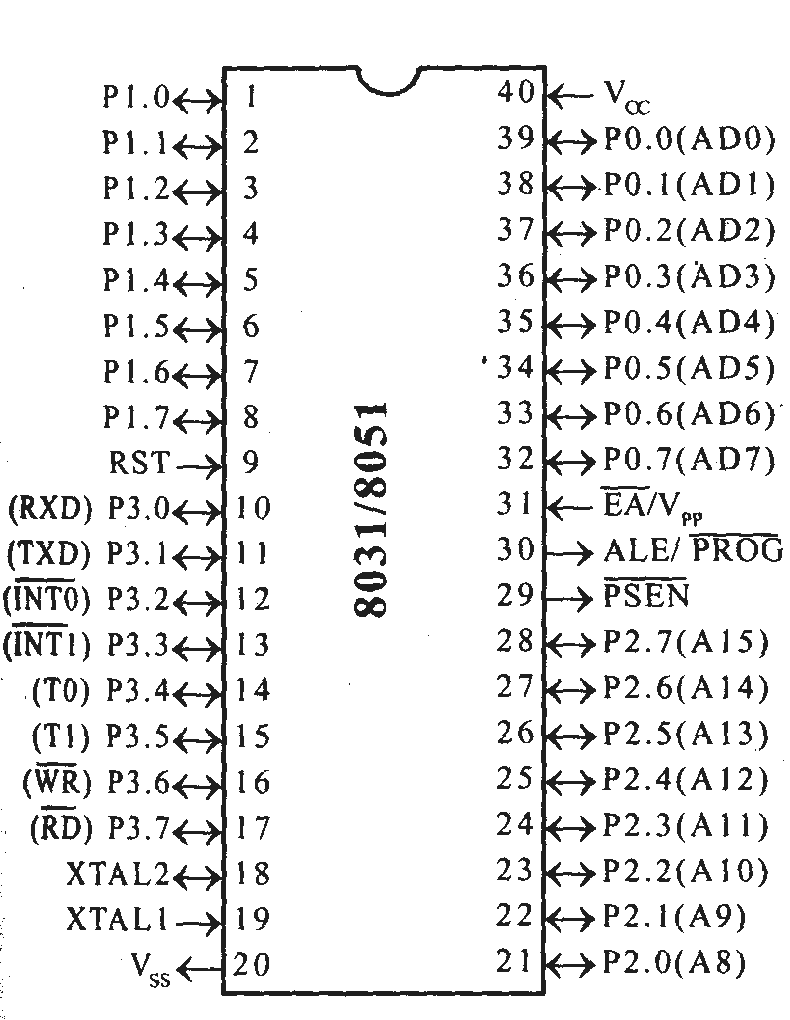
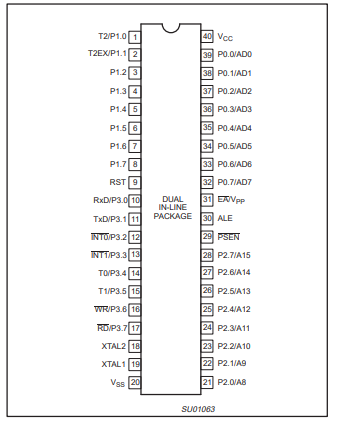
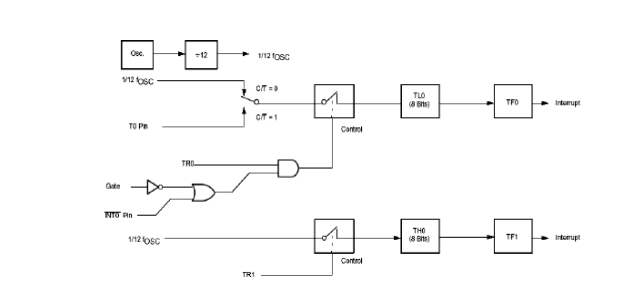
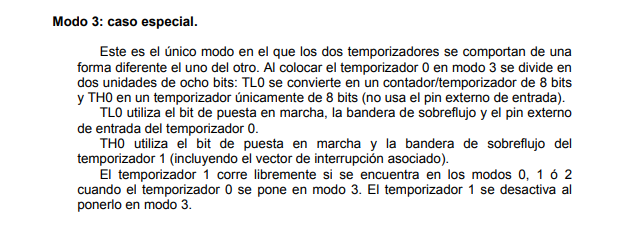
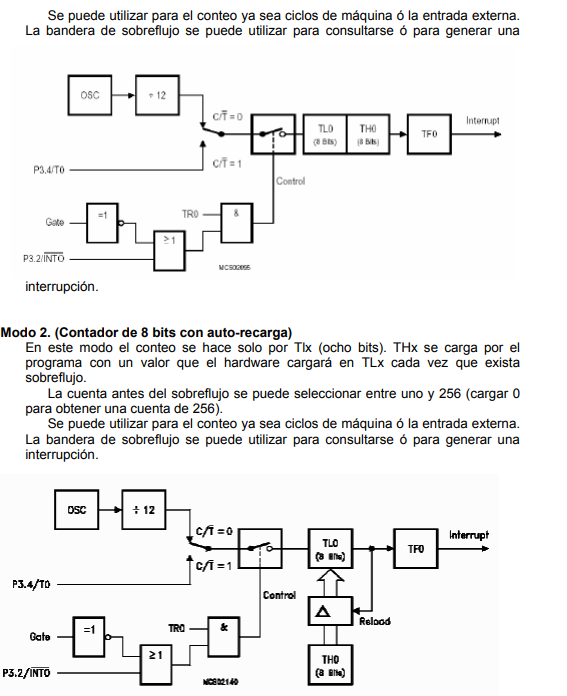
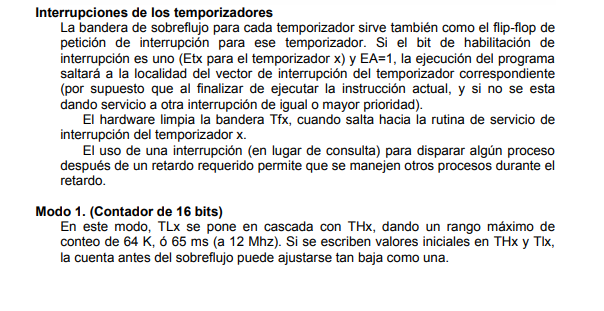
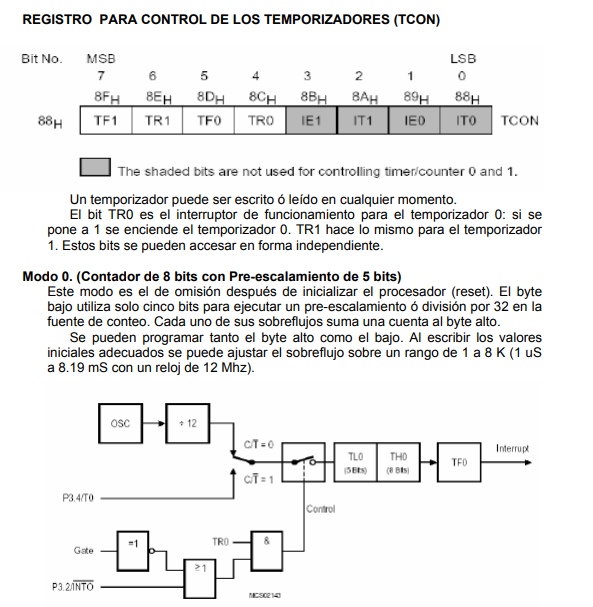
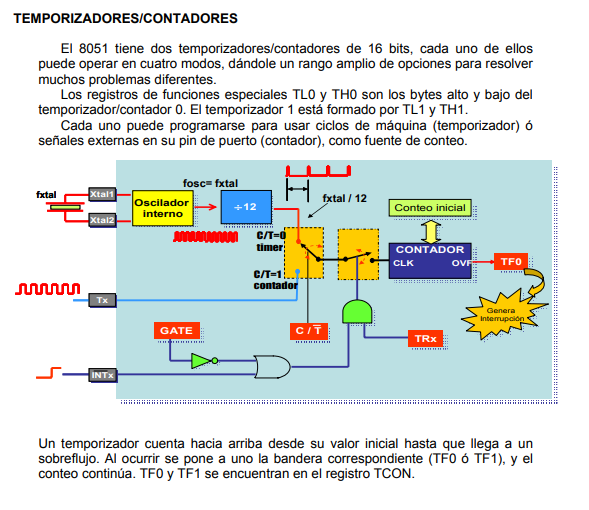
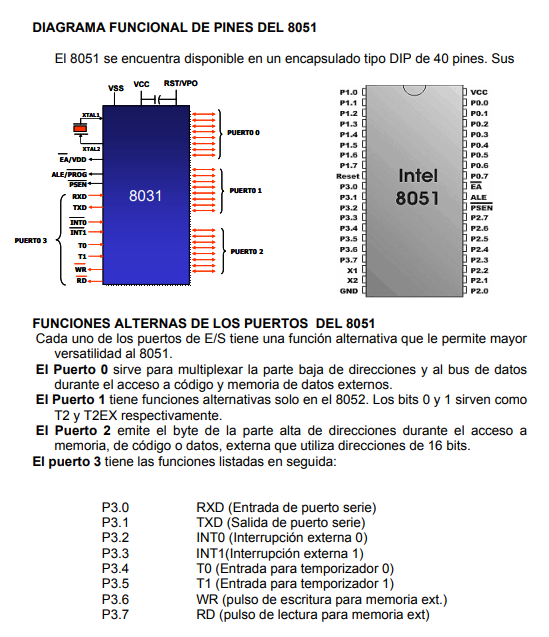


Diagrama de pines Intel 8052



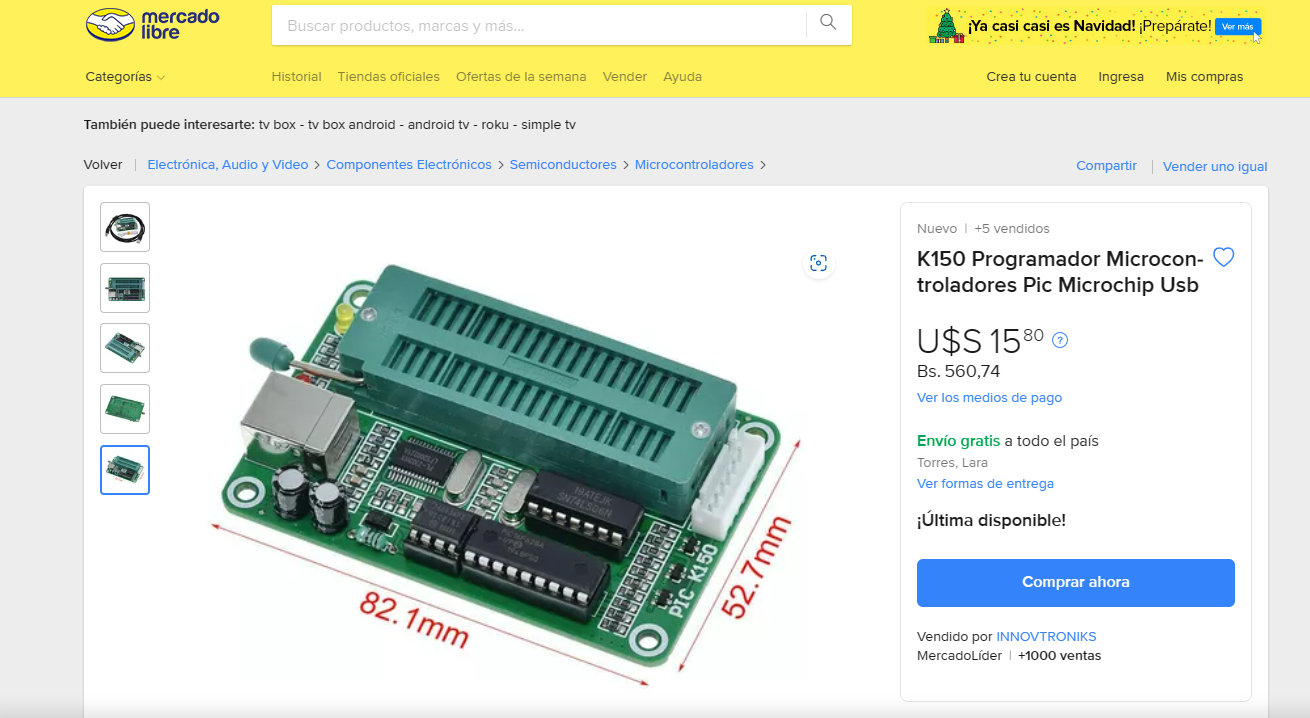
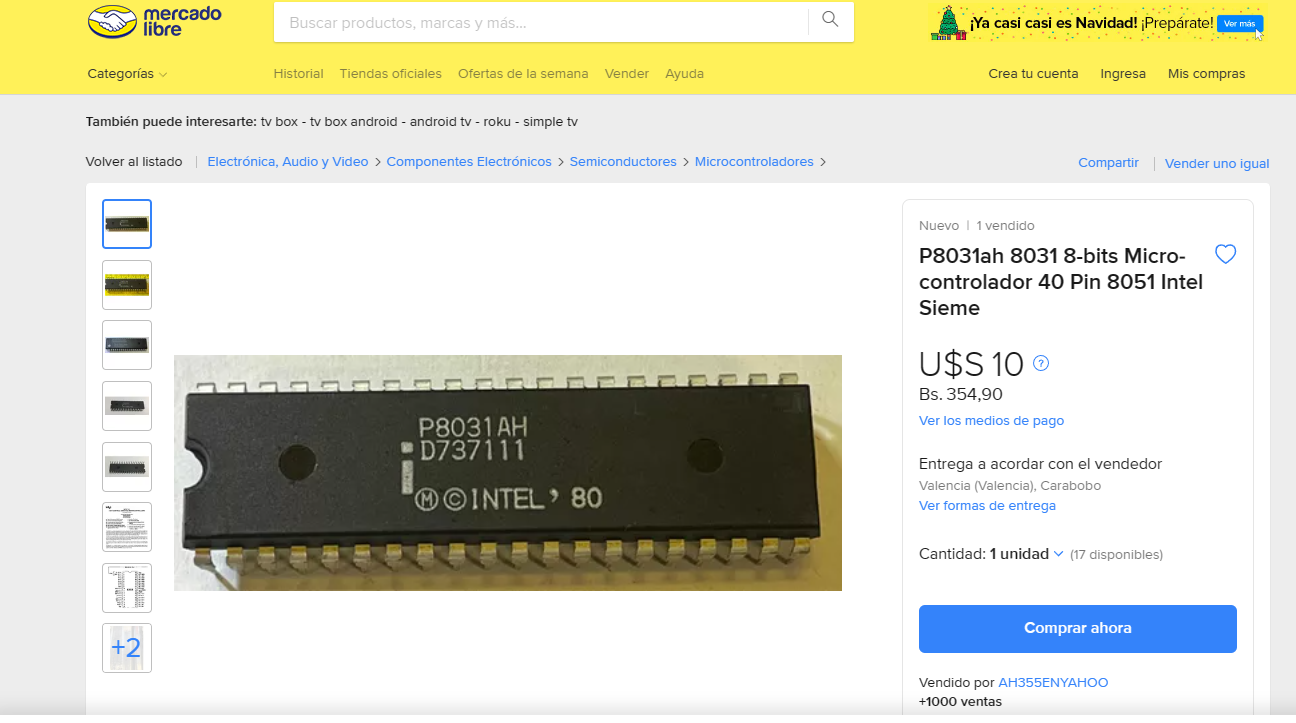
[Document: (nxp.com)](https://www.nxp.com/docs/en/data-sheet/8XC51_8XC52.pdf) <- fuente data sheet 80C51/52

Puertos y temporizadores familia Intel



cuanto cuesta el micro controlador y que necesito para programarlo

bueno 10$ el micro y 15$ la board 25$ sin contar la pc para programarlos nada barato .



#include <REG51.H> /\* Declaraciones de los SFR \*/

#include <stdio.h>/\* Declaraciones para funciones de E/S (como printf) \*/

main () {

SCON = 0x52; /\* puerto serial, modo 1 \*/

TMOD = 0x20; /\* temporizador 1, modo 2 \*/

TH1 = -13; /\* conteo de recarga para 2400 baudios \*/

TR1 = 1; /\* inicia temporizador 1 \*/

while (1) /\* repite infinitamente \*/

{

printf (“Hello World\n”); /\* Muestra “Hello world” \*/

}

}

Un hola mundo en un bucle infinito con inclusión de librería REG51.H pero REG51.H no es STD ansi c.

**PLC.**

Introducción

Un Controlador Lógico Programable, más conocido **PLC** (*Programmable Logic Controller*, debido a sus siglas en inglés) es básicamente una computadora que se utiliza en la ingeniería de automatización para las industrias, es decir, para el control de la maquinaria de una fábrica o de situaciones mecánicas.

Se trata de dispositivos electrónicos programables que se pueden adaptar a las necesidades de tu compañía o fábrica, sobre todo en las líneas de producción. Existen diferentes proveedores que ayudarán a programar cada uno de estos dispositivos, con el objetivo de que funcione correctamente, pero, sobre todo, que esté personalizado para el uso de tu empresa.

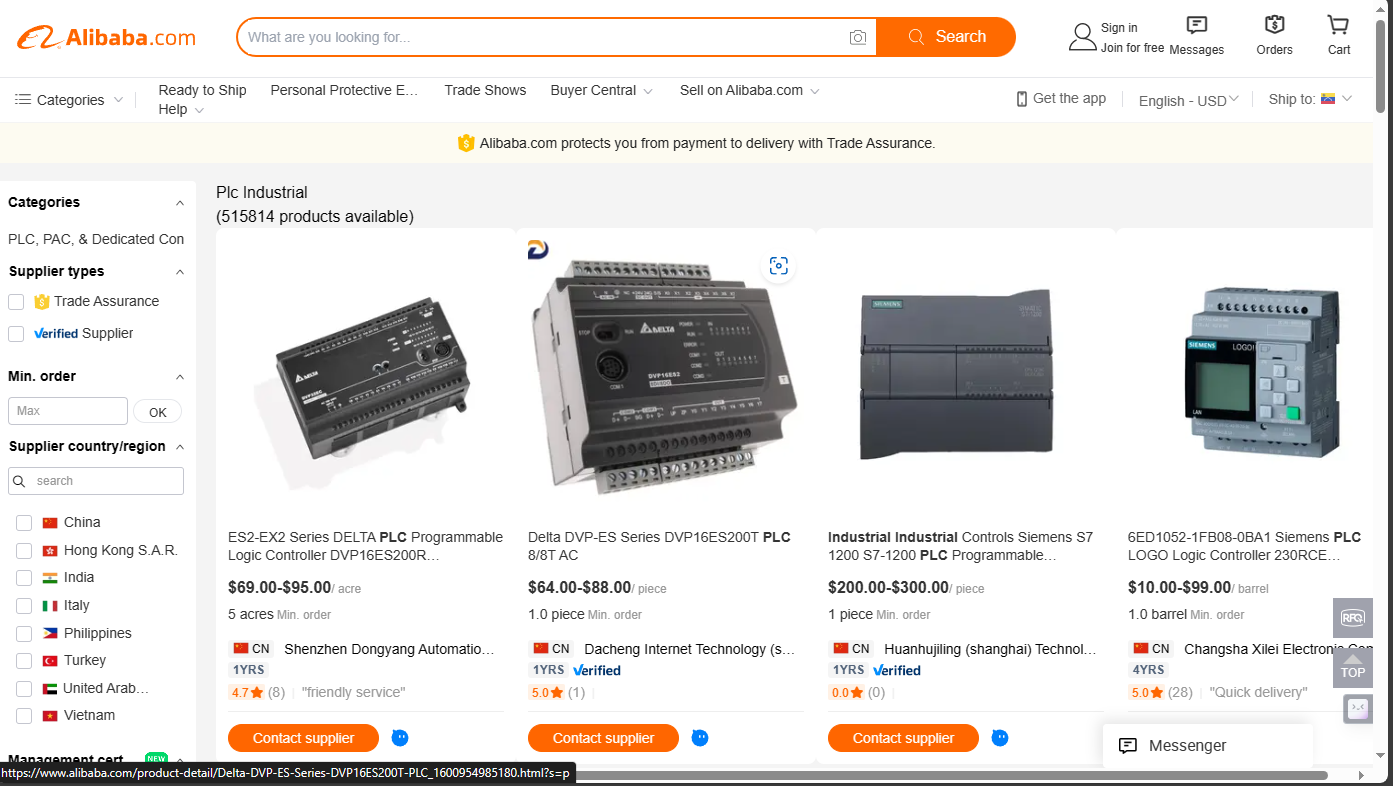
También se le conoce como cerebro electrónico, encargado de accionar a otros componentes de maquinaria para que realicen acciones que pudieran ser peligrosas para los seres humanos o muy lentas si se hace manualmente.

Actualmente se usan para aplicaciones industriales, aunque ya se están viendo casos en los que se aplican para usos domésticos o comerciales.

Aplicaciones de los PLC

Básicamente los **PLC** resuelven requerimientos de control de procesos y secuencias de la maquinaria dentro del sector industrias.

Precio de un plc según alibaba



Aprox 100 $ para el siemens 6ED1052

Programación de un plc

¿Qué es la programación de PLC?

La [programación del controlador lógico programable](https://www.chemik.es/blog/programacion-plc-automatizacion/), generalmente conocida por programación PLC, es un **sistema de revisión industrial** que identifica la información acerca de los dispositivos de entrada y realiza tareas industriales apoyadas en decisiones de control de máquinas y dispositivos de salida del cliente.

Abarca actividades importantes de control de dispositivos de salida para obtener información necesaria según las demandas del cliente.

Según los tipos de programación en PLC, el controlador lógico programable puede encontrarse en forma textual, gráfica o por una interfaz para representar la lógica. El software de la programación de PLC puede ser descargada en el hardware de la programación de PLC de los fabricantes que permitan este tipo de programación.

Lenguaje de programación: tipos de programación en PLC

Uno de los **aspectos diferenciales** de los distintos tipos de programación en PLC es el **lenguaje de programación** que se utiliza en su instalación. El sistema se completa utilizando uno de los 5 lenguajes para tipos de programación en PLC, los cuales detallamos en las siguientes líneas.

Los lenguajes de programación se diferencian a su vez de los símbolos, caracteres o reglas que utilizan según su diseño para la comunicación de los usuarios con las máquinas.

En la actualidad, el[**IEC 6131**](http://www.infoplc.net/files/documentacion/estandar_programacion/infoPLC_net_Intro_estandar_IEC_61131-3.pdf) es el lenguaje de programación en PLC que define los próximos tipos.

Lista de instrucciones (IL)

Este tipo de lenguaje de programación de PLC **se basa en el texto**, que funciona como el lenguaje básico. Su último propósito es utilizar un dispositivo que consiga procesar las instrucciones mediante los códigos AND, OR y LD.

Es el lenguaje de texto con más años de uso, siendo la base del resto de lenguajes y al que se recurría cuando los ordenadores no tenían capacidad gráfica.

Texto estructurado (ST)

El texto estructurado es otro de los lenguajes de programación de PLC, pero en este caso es usado **para ejecutar tareas complejas**, pues puede emplear las funciones matemáticas con algoritmos para realizar cualquier actividad repetitiva.

Se define en base a las funciones de entrada y de salida y está compuesto por un conjunto de instrucciones que se pueden ejecutar de manera condicionada.

Diagramas de bloques de funciones (FBD)

Este tipo de lenguaje de programación de PLC **se basa en gráficos**, utilizando bloques de símbolo lógico o un diagrama en el que se representan las funciones de los comandos de entrada y salida.

Lógica de escalera (LD)

Este lenguaje de lógica de escalera **se inspiró en función de la lógica del relé**. Se sirve principalmente en los interruptores y relé mecánico para controlar los procesos de estas industrias.

Un sistema lógico interno es el que dirige todo y reemplaza a las máquinas comunes físicas y que necesitan señales eléctricas para mantenerse activas.

Utiliza el lenguaje gráfico, evolución del lenguaje de la lista de instrucciones (IL). Se compone de dos niveles verticales de alimentación y de otros dos horizontales. Las instrucciones están ubicadas en el lado izquierdo y las salidas en el lado derecho. De esta forma, interpretará los datos de abajo arriba y de izquierda a derecha.

Cuadros de funciones secuenciales

Este último tipo de lenguaje de programación PLC **utiliza el gráfico de funciones** para abarcar todas sus misiones. Se beneficia de la transición, siendo esta un conjunto de instrucciones empleadas para automatizar los procesos, cuyo objetivo es realizar la tarea específica a través de las situaciones predefinidas.

Tipos de programación en PLC

Se pueden **diferenciar** los distintos tipos de programación en PLC según los **diversos tamaños y estructura**, los cuales se dividen en dos grupos.

* **PLC Compacto:** este tipo de sistema entra dentro de los tipos de programación en PLC, el cual abarca los programas fijos de la unidad E/S y de su beneficiario principal.
* **PLC Modular:** este tipo de programación tiene muchas unidades de E/S y se pueden fabricar juntas, por lo que personaliza el servicio para los fines de entrada y salida deseados por la industria.

Aplicaciones de los plc.

Esta tecnología de implementación sencilla y ejecución rápida, nos permite automatizar los procesos de nuestro proyecto con un bajo mantenimiento e instalación personalizada. Como mencionamos anteriormente, la función de estos ordenadores es el almacenamiento de datos, útiles para dos funciones: el control y testeo de las piezas que conforman los sistemas utilizados en la industria y la automatización de procesos industriales mediante la computarización de órdenes y funcionamientos específicos de cada máquina.

Se ocupan en el sector industrial, debido a que resuelven requerimientos de control de procesos y secuencias de la maquinaria; algunos ejemplos para los que se aplican estos controladores en la industria moderna son:

**Maquinaria**

* • Máquinas de procesado de gravas, cementos y arenas.
* • Máquinas industriales para la madera y los muebles.
* • Maquinaria industrial del plástico.
* • Herramientas complejas.
* • Máquinas de ensamblaje.

**Instalaciones**

* Instalaciones de seguridad.
* Instalaciones de calefacción y aire acondicionado.
* Instalaciones de plantas para el embotellado.
* Instalaciones de transporte y almacenaje.
* Instalaciones para tratamientos térmicos.
* Instalaciones industriales azucareras.

**Industria automotriz**

* • Aplicaciones en cadenas de montaje para soldaduras, cabinas de pintura, ensamblaje, etc.
* • Uso en máquinas de herramientas como fresadoras, taladradoras, tornos, etc.
* • Fabricación de neumáticos.

**Plantas químicas**

* • Oleoductos, refinados, baños electrolíticos, tratamientos de aguas residuales y fecales, etc.
* • Control de procesos como el pesaje, la dosificación, la mezcla, etc.

**Sistemas de transporte automatizado**

En la logística y el transporte, los PLCs se utilizan para controlar sistemas de transporte automatizados, como transportadores y robots, en almacenes y centros de distribución.

**Industria alimentaria**

En la industria alimentaria, los PLCs son fundamentales para controlar procesos de producción, llenado y envasado, y para garantizar el cumplimiento de estándares de calidad y seguridad alimentaria.

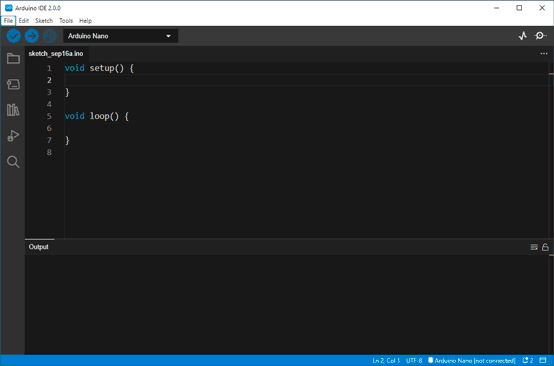
Arduino

Arduino [ha anunciado en su blog](https://blog.arduino.cc/2022/09/14/its-here-please-welcome-arduino-ide-2-0/) que, finalmente, ya se encuentra disponible la versión 2.0 de su entorno de desarrollo (IDE). Este incorpora numerosas mejoras respecto a la versión anterior.

La versión 2.0 estaba disponible como beta desde marzo de 2021. Gracias al esfuerzo de muchas personas que han probado esta nueva versión, hoy está disponible como versión estable pública.

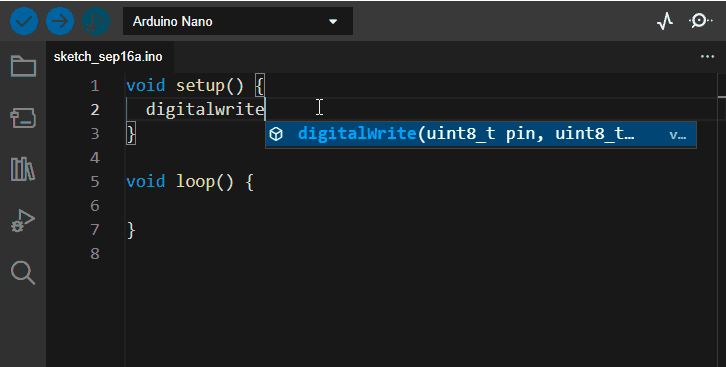
El nuevo IDE incorpora importantes mejoras tanto estéticas como funcionales, representando una importante mejora respecto a la versión anterior. Estas lo acercan a las características que esperamos en cualquier entorno de programación actual.

Está basado en Eclipse Theia framework, un proyecto Open Source para desarrollar IDE’s Desktop y Cloud. El front-end está escrito en TypeScript, mientras que la mayor parte del backend está desarrollado en Golang.



Eclipse Theia está diseñado para proporcionar una experiencia similar a la de VS Code, por lo que encontraremos que visualmente es muy similar. Así encontramos una importante mejora estética, con un UI modernizado y adaptativo, con soporte para themes.

Entre las mejoras más importantes encontramos la función de autocompletado, poder ir a la definición de una función (peek definition), el formato automático, o renombrado de variables y métodos. Estas son funciones muy útiles y habituales en casi cualquier IDE, y resultaba difícil de entender que el IDE de Arduino careciera de ellas.



Otro punto fuerte de la nueva versión es una importante mejora en la velocidad de compilación. Los tiempos se han reducido notable, especialmente al recompilar un programa.

Entre las nuevas funciones también se incluyen un mejorado Serial Plotter, copiar en formato markdown, o la capacidad de auto-actualización, por lo que ya no será necesario instalar nuevas versiones. También dispone integración con la Arduino Cloud, aunque personalmente no le encuentro demasiado interés, frente a otras alternativas como Git.

En definitiva, estas nuevas características son unas mejoras muy importantes que, quizás, llegan algo tarde. Especialmente si tenemos en cuenta que existen distintas alternativas, siendo el rival a batir la combinación Visual Studio Code + PlatformIO (como vimos [en esta entrada](https://www.luisllamas.es/como-programar-arduino-con-visual-studio-code-y-plaftormio/)).

Entre las carencias que más echo en falta está el subrayado de errores, permitir gestionar de forma más sencilla los proyectos con más de un archivo, o mantener el sistema de librerías global actual, en lugar añadidas de forma individual en el proyecto.

Sin embargo, pese a estos puntos de mejora y aunque existen mejores alternativas, no hay que olvidar que el punto fuerte del IDE de Arduino es su sencillez de uso. Así, está diseñado para usuarios causales o en iniciación.

De esta forma, un gran número de usuarios emplearán el entorno estándar de Arduino, mientras que otras alternativas como visual Studio code más PlatformIO quedan reservadas para usuarios más avanzados.

En este caso los usuarios del entorno estándar, que son la mayoría, van a encontrar una gran mejoría en la nueva versión 2.0. Esta supone una actualización muy de agradecer en un IDE que empezaba a notar el paso del tiempo.

**Ejemplos de código**

**Salida digital**

En este ejemplo el LED conectado al pin 13 parpadea cada segundo.

int ledPin = 13; // LED que se encuentra en el pin 13

void setup(){

pinMode(ledPin, OUTPUT); // El p1n 13 será una salida digital

}

void loop(){

digitalWrite(ledPin, HIGH); // Enciende el LED

delay(1000); // Pausa de 1 segundo

digitalWrite(ledPin, LOW); // Apaga el LED

delay(1000); // Pausa de 1 segundo

}

**Salida digital II**

En este ejemplo el LED conectado al pin 13 parpadea en un intervalo de tiempo variable que depende del número de veces que se ejecuta el programa (función *loop*)

int ledPin = 13; // LED que se encuentra en el pin 13

int n = 0; //Entero que contará el paso por la función loop

void setup(){

pinMode(ledPin, OUTPUT); // El p1n 13 será una salida digital

}

void loop(){

digitalWrite(ledPin, HIGH); // Enciende el LED

delay(1000); // Pausa de 1 segundo

digitalWrite(ledPin, LOW); // Apaga el LED

n++; //Incrementamos n

delay(delayVal(n)); //Pausa de un tiempo variable

}

//Función que devuelve un valor tipo entero según el parámetro pasado

int delayVal(int f){

return f\*100;

}

Cuanto cuesta arduino y que necesitamos para programar según mercado libre 9 $ aprox

