

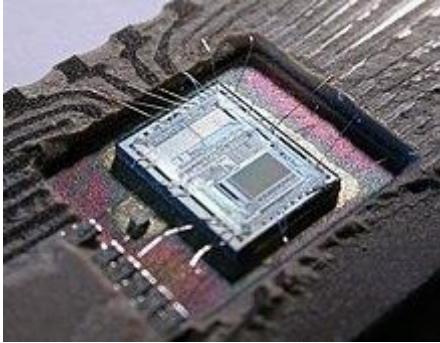
Microcontroladores:

Características y aplicaciones generales 1. Definición Recibe el nombre de controlador el dispositivo que se emplea para el gobierno de uno o varios procesos. Por ejemplo, el controlador que regula el funcionamiento de un horno dispone de un sensor que mide constantemente su temperatura interna y, cuando traspasa los límites prefijados, genera las señales adecuadas que accionan los actuadores que intentan llevar el valor de la temperatura dentro del rango estipulado. Aunque el concepto de controlador ha permanecido invariable a través del tiempo, su implementación física ha variado frecuentemente. Hace tres décadas los controladores se construían exclusivamente con componentes de lógica discreta; posteriormente se emplearon los microprocesadores, que se rodeaban con chips de memoria y E/S sobre una tarjeta de circuito impreso. En la actualidad, todos los elementos del controlador se han podido incluir en un chip, el cual recibe el nombre de microcontrolador. Realmente consiste en un sencillo pero completo computador contenido en el corazón (chip) de un circuito integrado.

Microcontrolador



Microcontrolador [Motorola 68HC11](#) y chips de soporte



Die del microcontrolador de [8 bits Intel 8742](#), con [CPU](#) a 12 [MHz](#), 128 [bytes](#) de [memoria RAM](#), 2048 bytes de [EPROM](#), y [E/S](#) en



un chip [Microcontrolador PIC 18F8720](#) en



encapsulado [TQFP](#) de 80 pines
TMS 1000 de Texas Instruments

Microcontrolador



Un **microcontrolador** (abreviado μC , **UC** o **mCU**) es un [circuito integrado](#) programable, capaz de ejecutar las órdenes grabadas en su memoria.

Está compuesto de varios bloques funcionales que cumplen una tarea específica. Un microcontrolador incluye en su interior las tres principales unidades funcionales de una [computadora](#): [unidad central](#) de [procesamiento](#), [memoria](#) y [periféricos](#) de [entrada/salida](#).

Algunos microcontroladores pueden utilizar palabras de cuatro bits y funcionan a velocidad de reloj con frecuencias tan bajas como 4 kHz, con un consumo de baja potencia ([mW o microwatts](#)). Por lo general, tendrá la capacidad de mantenerse a la espera de un evento como pulsar un botón o de otra interrupción; así, el consumo de energía durante el estado de reposo (reloj de la CPU y los periféricos de la mayoría) puede ser solo de nanowatts, lo que hace que muchos de ellos sean muy adecuados para aplicaciones con batería de larga duración. Otros microcontroladores pueden servir para roles de rendimiento crítico, donde sea necesario actuar más como un [procesador digital de señal](#) (DSP), con velocidades de reloj y consumo de energía más altos.

Cuando es fabricado el microcontrolador, no contiene datos en la memoria ROM. Para que pueda controlar algún proceso es necesario generar o crear y luego grabar en la [EEPROM](#) o equivalente del microcontrolador algún programa, el cual puede ser escrito en [lenguaje ensamblador](#) u otro lenguaje para microcontroladores; sin embargo, para que el programa pueda ser grabado en la memoria del microcontrolador, debe ser codificado en [sistema numérico hexadecimal](#) que es finalmente el sistema que hace trabajar al microcontrolador cuando este es alimentado con el voltaje adecuado y asociado a dispositivos [analógicos](#) y [discretos](#) para su funcionamiento.¹

Historia

El primer [microprocesador](#) fue el [Intel 4004](#) de [4 bits](#), lanzado en 1971, seguido por el [Intel 8008](#) y otros más capaces. Sin embargo, ambos procesadores requieren circuitos adicionales para implementar un sistema de trabajo, elevando el costo del sistema total.

El [Instituto Smithsonian](#) dice que los ingenieros de [Texas Instruments](#) [Gary Boone](#) y [Michael Cochran](#) lograron crear el primer microcontrolador, [TMS 1000](#), en 1971; fue comercializado en 1974. Combina [memoria ROM](#), [memoria RAM](#), [microprocesador](#) y reloj en un chip y estaba destinada a los [sistemas embebidos](#).²

Debido en parte a la existencia del TMS 1000,³ Intel desarrolló un sistema de ordenador en un chip optimizado para aplicaciones de control, el [Intel 8048](#), que comenzó a comercializarse en 1977.³ Combina memoria RAM y ROM en el mismo chip y puede encontrarse en más de mil millones de teclados de [compatible IBM PC](#), y otras numerosas aplicaciones. El en ese momento presidente de Intel, Luke J. Valenter, declaró que el microcontrolador es uno de los productos más exitosos en la historia de la compañía, y amplió el presupuesto de la división en más del 25%.

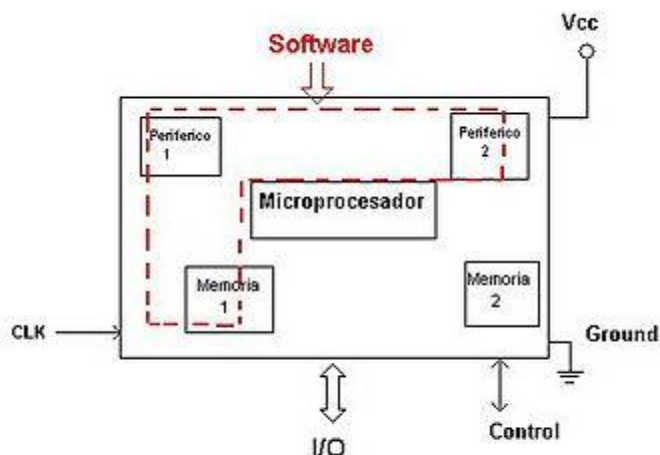
La mayoría de los microcontroladores en aquel momento tenían dos variantes. Unos tenían una memoria [EPROM](#) reprogramable, significativamente más caros que la variante [PROM](#) que era solo una vez programable. Para borrar la EPROM necesita exponer a la luz ultravioleta la tapa de cuarzo transparente. Los chips con todo opaco representaban un coste menor.

En 1993, el lanzamiento de la [EEPROM](#) en los microcontroladores (comenzando con el Microchip [PIC16x84](#))⁴ permite borrarla eléctrica y rápidamente sin necesidad de un paquete costoso como se requiere en [EPROM](#), lo que permite tanto la creación rápida de prototipos y la [programación en el sistema](#). El mismo año, [Atmel](#) lanza el primer microcontrolador que utiliza [memoria flash](#).⁵ Otras compañías rápidamente siguieron el ejemplo, con los dos tipos de memoria.

El costo se ha desplomado en el tiempo, con el más barato microcontrolador de 8 bits disponible por menos de 0,25 [dólares](#) para miles de unidades en 2009, y algunos microcontroladores de 32 bits a 1 dólar por cantidades similares. En la actualidad los microcontroladores son baratos y fácilmente disponibles para los aficionados, con grandes comunidades en línea para ciertos procesadores.

En el futuro, la [MRAM](#) podría ser utilizada en microcontroladores, ya que tiene resistencia infinita y el coste de su oblea semiconductora es relativamente bajo.

Características



Esquema de un microcontrolador. En esta figura, vemos al microcontrolador metido dentro de un encapsulado de circuito integrado, con su procesador (CPU), buses, memoria, periféricos y puertos de [entrada/salida](#). Fuera del encapsulado se ubican otros circuitos para completar periféricos internos y dispositivos que pueden conectarse a los pines de entrada/salida. También se conectarán a los pines del encapsulado la alimentación, masa, circuito de completamiento del oscilador y otros circuitos necesarios para que el microcontrolador pueda trabajar.

Los microcontroladores están diseñados para reducir el costo económico y el consumo de energía de un sistema en particular. Por eso el tamaño de la [unidad central de procesamiento](#), la cantidad de memoria y los periféricos incluidos dependerán de la aplicación. El control de un electrodoméstico sencillo como una batidora utilizará un procesador muy pequeño (4 u 8 bits) porque sustituirá a un autómata finito. En cambio, un reproductor de música o vídeo digital ([MP3](#) o [MP4](#)) requerirá de un procesador de [32 bits](#) o de [64 bits](#) y de uno o más [códecs](#) de [señal digital](#) (audio o vídeo). El control de un sistema de frenos ABS ([Antilock Brake System](#)) se basa normalmente en un microcontrolador de [16 bits](#), al igual que el sistema de control electrónico del motor en un [automóvil](#).

Los microcontroladores representan la inmensa mayoría de los chips de computadoras vendidos, sobre un 50% son controladores "simples" y el restante corresponde a [DSP](#) más especializados. Mientras se pueden tener uno o dos microprocesadores de propósito general en casa (Ud. está usando uno para esto), usted tiene distribuidos seguramente entre los electrodomésticos de su hogar una



o dos docenas de microcontroladores. Pueden encontrarse en casi cualquier dispositivo electrónico como [automóviles](#), [lavadoras](#), [hornos microondas](#), [teléfonos](#), etc.

Un microcontrolador difiere de una [unidad central de procesamiento](#) normal, debido a que es más fácil convertirla en una computadora en funcionamiento, con un mínimo de [circuitos integrados externos de apoyo](#). La idea es que el circuito integrado se coloque en el dispositivo, enganchado a la fuente de energía y de información que necesite, y eso es todo. Un microprocesador tradicional no le permitirá hacer esto, ya que espera que todas estas tareas sean manejadas por otros chips. Hay que agregarle los módulos de entrada y salida (puertos) y la memoria para almacenamiento de información.

Un microcontrolador típico tendrá un generador de reloj integrado y una pequeña cantidad de [memoria de acceso aleatorio](#) o [ROM/EPROM/EEPROM/flash](#), con lo que para hacerlo funcionar todo lo que se necesita son unos pocos programas de control y un cristal de sincronización. Los microcontroladores disponen generalmente también de una gran variedad de [dispositivos de entrada/salida](#), como [convertidor analógico digital](#), [temporizadores](#), [UARTs](#) y buses de interfaz serie especializados, como [I²C](#) y [CAN](#). Frecuentemente, estos dispositivos integrados pueden ser controlados por instrucciones de procesadores especializados. Los modernos microcontroladores frecuentemente incluyen un lenguaje de programación integrado, como el lenguaje de programación [BASIC](#) que se utiliza bastante con este propósito.

Los microcontroladores negocian la velocidad y la flexibilidad para facilitar su uso. Debido a que se utiliza bastante sitio en el chip para incluir funcionalidad, como los [dispositivos de entrada/salida](#) o la [memoria](#) que incluye el microcontrolador, se ha de prescindir de cualquier otra circuitería.

Arquitecturas de computadora

Básicamente existen dos arquitecturas de computadoras, y por supuesto, están presentes en el mundo de los microcontroladores: Von Neumann y Harvard. Ambas se diferencian en la forma de conexión de la memoria al procesador y en los buses que cada una necesita.

Arquitectura Von Neumann[

La [arquitectura Von Neumann](#) utiliza el mismo dispositivo de almacenamiento tanto para las instrucciones como para los datos, siendo la que se utiliza en un [ordenador personal](#) porque permite ahorrar una buena cantidad de líneas de E/S, que son bastante costosas, sobre todo para aquellos sistemas donde el procesador se monta en algún tipo de [zócalo](#) alojado en una [placa madre](#). También esta organización les ahorra a los diseñadores de placas madre una buena cantidad de problemas y reduce el costo de este tipo de sistemas.

En un ordenador personal, cuando se carga un programa en memoria, a este se le asigna un espacio de direcciones de la memoria que se divide en segmentos, de los cuales típicamente tendremos los siguientes: código (programa), datos y [pila](#). Es por ello que podemos hablar de la memoria como un todo, aunque existan distintos dispositivos físicos en el sistema ([disco duro](#), [memoria RAM](#), [memoria flash](#), [unidad de disco óptico](#)...).

En el caso de los microcontroladores, existen dos tipos de memoria bien definidas: memoria de datos (típicamente algún tipo de [SRAM](#)) y memoria de programas ([ROM](#), [PROM](#), EEPROM, flash u de otro tipo no volátil). En este caso la organización es distinta a las del ordenador personal, porque hay circuitos distintos para cada memoria y normalmente no se utilizan los registros de segmentos, sino que la memoria está segregada y el acceso a cada tipo de memoria depende de las instrucciones del procesador.

A pesar de que en los sistemas integrados con arquitectura Von Neumann la memoria esté segregada, y existan diferencias con respecto a la definición tradicional de esta arquitectura; los buses para acceder a ambos tipos de memoria son los mismos, del procesador solamente salen el bus de datos, el de direcciones, y el de control. Como conclusión, la arquitectura no ha sido alterada, porque la forma en que se conecta la memoria al procesador sigue el mismo principio definido en la arquitectura básica.

Algunas familias de microcontroladores como la [Intel 8051](#) y la Z80 implementan este tipo de arquitectura, fundamentalmente porque era la utilizada cuando aparecieron los primeros microcontroladores.

Arquitectura Harvard

La otra variante es la [arquitectura Harvard](#), y por excelencia la utilizada en supercomputadoras, en los microcontroladores, y sistemas integrados en general. En este caso, además de la memoria, el procesador tiene los buses segregados,

de modo que cada tipo de memoria tiene un bus de datos, uno de direcciones y uno de control.

La ventaja fundamental de esta arquitectura es que permite adecuar el tamaño de los buses a las características de cada tipo de memoria; además, el procesador puede acceder a cada una de ellas de forma simultánea, lo que se traduce en un aumento significativo de la velocidad de procesamiento. Típicamente los sistemas con esta arquitectura pueden ser dos veces más rápidos que sistemas similares con arquitectura Von Neumann.

La desventaja está en que consume muchas líneas de E/S del procesador; por lo que en sistemas donde el procesador está ubicado en su propio encapsulado, solo se utiliza en supercomputadoras. Sin embargo, en los microcontroladores y otros sistemas integrados, donde usualmente la memoria de datos y programas comparten el mismo encapsulado que el procesador, este inconveniente deja de ser un problema serio y es por ello que encontramos la arquitectura Harvard en la mayoría de los microcontroladores.

Por eso es importante recordar que un microcontrolador se puede configurar de diferentes maneras, siempre y cuando se respete el tamaño de memoria que este requiera para su correcto funcionamiento.

Procesador en detalle

En los años 1970, la electrónica digital no estaba suficientemente desarrollada, pero dentro de la electrónica ya era una especialidad consagrada. En aquel entonces las computadoras se diseñaban para que realizaran algunas operaciones muy simples, y si se quería que estas máquinas pudiesen hacer cosas diferentes, era necesario realizar cambios bastante significativos al hardware.

A principios de la década de 1970, una empresa japonesa le encargó a una joven compañía norteamericana que desarrollara un conjunto de circuitos para producir una calculadora de bajo costo. [Intel](#) se dedicó de lleno a la tarea y entre los circuitos encargados desarrolló uno muy especial, algo no creado hasta la fecha: el primer microprocesador integrado.

El [Intel 4004](#) salió al mercado en 1971, es una máquina digital sincrónica compleja, como cualquier otro circuito lógico secuencial sincrónico. Sin embargo, la ventaja de este componente está en que aloja internamente un conjunto de circuitos digitales que pueden hacer operaciones corrientes para el cálculo y procesamiento de datos, pero desde una óptica diferente: sus entradas son una serie de códigos bien definidos, que permiten hacer operaciones de carácter



específico cuyo resultado está determinado por el tipo de operación y los operandos involucrados.

Visto así, no hay nada de especial en un microprocesador; la maravilla está en que la combinación adecuada de los códigos de entrada, su ejecución secuencial, el poder saltar hacia atrás o adelante en la secuencia de códigos sobre la base de decisiones lógicas u órdenes específicas, permite que la máquina realice gran cantidad de operaciones complejas, no contempladas en los simples códigos básicos.

Hoy estamos acostumbrados a los sistemas con microprocesadores, pero en el *lejano* 1971 esta era una forma de pensar un poco diferente y hasta escandalosa, a tal punto que [Busicom](#), la empresa que encargó los chips a Intel, no se mostró interesada en el invento, por lo que Intel lo comercializó para otros que mostraron interés; el resto es historia: una revolución sin precedentes en el avance tecnológico de la humanidad.

Es lógico pensar que el invento del microprocesador integrado no fue una revelación divina para sus creadores, sino que se sustentó en los avances, existentes hasta el momento, en el campo de la electrónica digital y las teorías sobre computación. Pero sin lugar a dudas fue la gota que colmó la copa de la revolución científico-técnica, porque permitió desarrollar aplicaciones impensadas o acelerar algunas ya encaminadas.

Ahora comenzaremos a ver cómo es que está hecho un procesador, no será una explicación demasiado detallada porque desde su invención este ha tenido importantes revoluciones propias, pero hay aspectos básicos que no han cambiado y que constituyen la base de cualquier microprocesador. En la Figura 'Esquema de un microcontrolador' podemos ver la estructura típica de un microprocesador, con sus componentes fundamentales, claro está que ningún procesador real se ajusta exactamente a esta estructura, pero aun así nos permite conocer cada uno de sus elementos básicos y sus interrelaciones.

Registros

Son un espacio de memoria muy reducido pero necesario para cualquier microprocesador, de aquí se toman los datos para varias operaciones que debe realizar el resto de los circuitos del procesador. Los registros sirven para almacenar los resultados de la ejecución de instrucciones, cargar datos desde la memoria externa o almacenarlos en ella.

Aunque la importancia de los registros parezca trivial, no lo es en absoluto. De hecho una parte de los registros, la destinada a los datos, es la que determina uno

de los parámetros más importantes de cualquier microprocesador. Cuando escuchamos que un procesador es de 4, 8, 16, 32 o 64 bits, nos estamos refiriendo a procesadores que realizan sus operaciones con registros de datos de ese tamaño, y por supuesto, esto determina muchas de las potencialidades de estas máquinas.

Mientras mayor sea el número de bits de los registros de datos del procesador, mayores serán sus prestaciones, en cuanto a poder de cómputo y velocidad de ejecución, ya que este parámetro determina la potencia que se puede incorporar al resto de los componentes del sistema, por ejemplo, no tiene sentido tener una ALU de 16 bits en un procesador de 8 bits.

Por otro lado un procesador de 16 bits, puede que haga una suma de 16 bits en un solo ciclo de máquina, mientras que uno de 8 bits deberá ejecutar varias instrucciones antes de tener el resultado, aun cuando ambos procesadores tengan la misma velocidad de ejecución para sus instrucciones. El procesador de 16 bits será más rápido porque puede hacer el mismo tipo de tareas que uno de 8 bits, en menos tiempo.

Unidad de control

Esta unidad es de las más importantes en el procesador, en ella recae la lógica necesaria para la decodificación y ejecución de las instrucciones, el control de los registros, la ALU, los buses y cuanto cosa más se quiera meter en el procesador.

La unidad de control es uno de los elementos fundamentales que determinan las prestaciones del procesador, ya que su tipo y estructura determina parámetros tales como el tipo de [conjunto de instrucciones](#), velocidad de ejecución, tiempo del ciclo de máquina, tipo de buses que puede tener el sistema, manejo de interrupciones y un buen número de cosas más que en cualquier procesador van a parar a este bloque.

Por supuesto, las unidades de control son el elemento más complejo de un procesador y normalmente están divididas en unidades más pequeñas trabajando de conjunto. La unidad de control agrupa componentes tales como la unidad de decodificación, unidad de ejecución, controladores de memoria caché, controladores de buses, [controlador de interrupciones](#), [pipelines](#), entre otros elementos, dependiendo siempre del tipo de procesador.

Unidad aritmético-lógica (ALU)

Como los procesadores son circuitos que hacen básicamente operaciones lógicas y matemáticas, se le dedica a este proceso una unidad completa, con cierta independencia. Aquí es donde se realizan las sumas, restas, y operaciones lógicas típicas del [álgebra de Boole](#).

Actualmente este tipo de unidades ha evolucionado mucho y los procesadores más modernos tienen varias ALU, especializadas en la realización de operaciones complejas como las operaciones en [coma flotante](#). De hecho en muchos casos le han cambiado su nombre por el de «coprocesador matemático», aunque este es un término que surgió para dar nombre a un tipo especial de procesador que se conecta directamente al procesador más tradicional.

Su impacto en las prestaciones del procesador es también importante porque, dependiendo de su potencia, tareas más o menos complejas, pueden hacerse en tiempos muy cortos, como por ejemplo, los cálculos en coma flotante.

Buses

Son el medio de comunicación que utilizan los diferentes componentes del procesador para intercambiar información entre sí, finalmente los buses o una parte de ellos estarán reflejados en los pines del encapsulado del procesador.

En el caso de los microcontroladores, no es común que los buses estén reflejados en el encapsulado del circuito, ya que estos se destinan básicamente a las E/S de propósito general y periféricos del sistema.

Existen tres tipos de buses:

- Dirección: Se utiliza para seleccionar al dispositivo con el cual se quiere trabajar o en el caso de las memorias, seleccionar el dato que se desea leer o escribir.
- Datos: Se utiliza para mover los datos entre los dispositivos de *hardware* (entrada y salida).
- Control: Se utiliza para gestionar los distintos procesos de escritura lectura y controlar la operación de los dispositivos del sistema.

Conjunto de instrucciones

Aunque no aparezca en el esquema, no podíamos dejar al conjunto o repertorio de instrucciones fuera de la explicación, porque este elemento determina lo que puede hacer el procesador.

Define las operaciones básicas que puede realizar el procesador, que conjugadas y organizadas forman lo que conocemos como software. El [conjunto de instrucciones](#) vienen siendo como las letras del alfabeto, el elemento básico del lenguaje, que organizadas adecuadamente permiten escribir palabras, oraciones y cuanto programa se le ocurra.

Existen dos tipos básicos de repertorios de instrucciones, que determinan la arquitectura del procesador: CISC y RISC.

CISC, del inglés [Complex Instruction Set Computing](#), Computadora de conjunto de instrucciones complejo. Los microprocesadores CISC tienen un conjunto de instrucciones que se caracteriza por ser muy amplio y que permiten realizar operaciones complejas entre operandos situados en la memoria o en los registros internos. Este tipo de repertorio dificulta el paralelismo entre instrucciones, por lo que en la actualidad, la mayoría de los sistemas CISC de alto rendimiento convierten las instrucciones complejas en varias instrucciones simples del tipo RISC, llamadas generalmente microinstrucciones.

Dentro de los microcontroladores CISC podemos encontrar a la popular familia [Intel 8051](#) y la Z80, aunque actualmente existen versiones CISC-RISC de estos microcontroladores, que pretenden aprovechar las ventajas de los procesadores RISC a la vez que se mantiene la compatibilidad hacia atrás con las instrucciones de tipo CISC.

[RISC](#), del inglés *Reduced Instruction Set Computer*, Computadora con Conjunto de Instrucciones Reducido. Se centra en la obtención de procesadores con las siguientes características fundamentales:

- Instrucciones de tamaño fijo.
- Pocas instrucciones.
- Solo las instrucciones de carga y almacenamiento acceden a la memoria de datos.
- Número relativamente elevado de registros de propósito general.

Una de las características más destacables de este tipo de procesadores es que posibilitan el paralelismo en la ejecución, y reducen los accesos a memoria. Es por eso que los procesadores más modernos, tradicionalmente basados en

arquitecturas CISC, implementan mecanismos de traducción de instrucciones CISC a RISC, para aprovechar las ventajas de este tipo de procesadores.

Los procesadores de los microcontroladores PIC son de tipo RISC.

Memoria

Anteriormente se ha visto que la memoria en los microcontroladores debe estar ubicada dentro del mismo encapsulado, esto es así la mayoría de las veces, porque la idea fundamental es mantener el grueso de los circuitos del sistema dentro de un solo integrado.

En los microcontroladores la memoria no es abundante, aquí no encontrará Gigabytes de memoria como en las computadoras personales. Típicamente la memoria de programas no excederá de 16 K-localizaciones de memoria no volátil (flash o eeprom) para contener los programas.

La memoria RAM está destinada al almacenamiento de información temporal que será utilizada por el procesador para realizar cálculos u otro tipo de operaciones lógicas. En el espacio de direcciones de memoria RAM se ubican además los registros de trabajo del procesador y los de configuración y trabajo de los distintos periféricos del microcontrolador. Es por ello que en la mayoría de los casos, aunque se tenga un espacio de direcciones de un tamaño determinado, la cantidad de memoria RAM de que dispone el programador para almacenar sus datos es menor que la que puede direccionar el procesador.

El tipo de memoria utilizada en las memorias RAM de los microcontroladores es SRAM, lo que evita tener que implementar sistemas de [refrescamiento](#) como en el caso de las computadoras personales, que utilizan gran cantidad de memoria, típicamente alguna tecnología DRAM. A pesar de que la memoria SRAM es más costosa que la DRAM, es el tipo adecuado para los microcontroladores porque estos poseen pequeñas cantidades de memoria RAM.

En el caso de la memoria de programas se utilizan diferentes tecnologías, y el uso de una u otra depende de las características de la aplicación a desarrollar, a continuación se describen las cinco tecnologías existentes, que mayor utilización tienen o han tenido:

- [Memoria ROM](#) (*Read-Only Memory*) de máscara. En este caso no se “graba” el programa en memoria sino que el microcontrolador se fabrica con el programa, es un proceso similar al de producción de los CD comerciales mediante masterización. El costo inicial de producir un circuito de este tipo es alto, porque el diseño y producción de la máscara es un proceso costoso, sin

embargo, cuando se necesitan varios miles o incluso cientos de miles de microcontroladores para una aplicación determinada, como por ejemplo, algún electrodoméstico, el costo inicial de producción de la máscara y el de fabricación del circuito se distribuye entre todos los circuitos de la serie, y el costo final de esta es bastante menor que el de sus semejantes con otro tipo de memoria.

- Memoria PROM (*Programmable Read-Only Memory*) también conocida como OTP (*One Time Programmable*).

Los microcontroladores con memoria OTP se pueden programar una sola vez, con algún tipo de programador. Se utilizan en sistemas donde el programa no requiera futuras actualizaciones y para series relativamente pequeñas, donde la variante de máscara sea muy costosa, también para sistemas que requieren serialización de datos, almacenados como constantes en la memoria de programas.

- Memoria EPROM (*Erasable Programmable Read Only Memory*). Los microcontroladores con este tipo de memoria son muy fáciles de identificar porque su encapsulado es de cerámica y llevan encima una ventanita de vidrio desde la cual puede verse la oblea de silicio del microcontrolador.

Se fabrican así porque la memoria EPROM es reprogramable, pero antes debe borrarse, y para ello hay que exponerla a una fuente de luz ultravioleta, el proceso de grabación es similar al empleado para las memorias OTP.

Al aparecer tecnologías menos costosas y más flexibles, como las memorias EEPROM y FLASH, este tipo de memoria han caído en desuso, se utilizaban en sistemas que requieren actualizaciones del programa y para los procesos de desarrollo y puesta a punto.

- EEPROM (*Electrical Erasable Programmable Read Only Memory*). Fueron el sustituto natural de las memorias EPROM, la diferencia fundamental es que pueden ser borradas eléctricamente, por lo que la ventanilla de cristal de cuarzo y los encapsulados cerámicos no son necesarios.

Al disminuir los costos de los encapsulados, los microcontroladores con este tipo de memoria se hicieron más baratos y cómodos para trabajar que sus equivalentes con memoria EPROM.

Otra característica destacable de este tipo de microcontrolador es que fue en ellos donde comenzaron a utilizarse los sistemas de programación en el sistema que evitan tener que sacar el microcontrolador de la tarjeta que lo aloja para hacer actualizaciones al programa.

- [Memoria flash](#). En el campo de las memorias reprogramables para microcontroladores, son el último avance tecnológico en uso a gran escala, y han sustituido a los microcontroladores con memoria EEPROM.

A las ventajas de las memorias flash se le adicionan su gran densidad respecto a sus predecesoras lo que permite incrementar la cantidad de memoria de programas a un costo muy bajo. Pueden además ser programadas con las mismas tensiones de alimentación del microcontrolador, el acceso en lectura y la velocidad de programación es superior, disminución de los costos de producción, entre otras.

Lo más habitual es encontrar que la memoria de programas y datos está ubicada toda dentro del microcontrolador, de hecho, actualmente son pocos los microcontroladores que permiten conectar memoria de programas en el exterior del encapsulado. Las razones para estas “limitaciones” están dadas porque el objetivo fundamental es obtener la mayor integración posible y conectar memorias externas consume líneas de E/S que son uno de los recursos más preciados de los microcontroladores.

A pesar de lo anterior existen familias como la [Intel 8051](#) cuyos microcontroladores tienen la capacidad de ser expandidos en una variada gama de configuraciones para el uso de memoria de programas externa. En el caso de los PIC, estas posibilidades están limitadas solo a algunos microcontroladores de la gama alta, la Figura 5 muestra algunas de las configuraciones para memoria de programa que podemos encontrar en los microcontroladores. La configuración (a) es la típica y podemos encontrarla casi en el 100% de los microcontroladores. La configuración (b) es poco frecuente y generalmente se logra configurando al microcontrolador para sacrificar la memoria de programas interna, sin embargo el [Intel 8031](#) es un microcontrolador sin memoria de programas interna. La configuración (c) es la que se encuentra habitualmente en los microcontroladores que tienen posibilidades de expandir su memoria de programas como algunos PIC de gama alta.

Cuando se requiere aumentar la cantidad de memoria de datos, lo más frecuente es colocar dispositivos de memoria externa en forma de periféricos, de esta forma se pueden utilizar memorias RAM, FLASH o incluso discos duros como

los de los ordenadores personales, mientras que para los cálculos y demás operaciones que requieran almacenamiento temporal de datos se utiliza la memoria RAM interna del microcontrolador. Esta forma de expandir la memoria de datos está determinada, en la mayoría de los casos, por el tipo de repertorio de instrucciones del procesador y porque permite un elevado número de configuraciones distintas, además del consiguiente ahorro de líneas de E/S que se logra con el uso de memorias con buses de comunicación serie.

Interrupciones

Las interrupciones son esencialmente llamadas a [subrutina](#) generadas por los dispositivos físicos, al contrario de las subrutinas normales de un programa en ejecución. Como el salto de subrutina no es parte del hilo o secuencia de ejecución programada, el controlador guarda el estado del procesador en la [pila](#) de memoria y entra a ejecutar un código especial llamado "manejador de interrupciones" que atiende al periférico específico que generó la interrupción. Al terminar la rutina, una instrucción especial le indica al procesador el fin de la atención de la interrupción. En ese momento el controlador restablece el estado anterior, y el programa que se estaba ejecutando antes de la interrupción sigue como si nada hubiese pasado. Las rutinas de atención de interrupciones deben ser lo más breves posibles para que el rendimiento del sistema sea satisfactorio, porque normalmente cuando una interrupción es atendida, todas las demás interrupciones están en espera.

Imagine que está esperando la visita de un amigo, al que llamaremos Juan. Usted y Juan han acordado que cuando él llegue a su casa esperará pacientemente a que le abra la puerta. Juan no debe tocar a la puerta porque alguien en la casa duerme y no quiere que le despierten.

Ahora usted ha decidido leer un libro mientras espera a que Juan llegue a la casa, y para comprobar si ha llegado, cada cierto tiempo detiene la lectura, marca la página donde se quedó, se levanta y va hasta la puerta, abre y comprueba si

Juan ha llegado, si éste todavía no está en la puerta, esperará unos minutos, cerrará la puerta y regresará a su lectura durante algún tiempo.

Como verá este es un método poco eficiente para esperar a Juan porque requiere que deje la lectura cada cierto tiempo y vaya hasta la puerta a comprobar si él ha llegado, además debe esperar un rato si todavía no llega. Y por si fuera poco, imagine que Juan no llega nunca porque se le presentó un problema, tuvo que cancelar la cita y no pudo avisarle a tiempo, o peor, que Juan ha llegado a la puerta un instante después que usted la cerraba. Juan, respetando lo acordado, espera un tiempo, pero se cansa de esperar a que le abran y decide marcharse porque cree que ya usted no está en la casa o no puede atenderlo. A este método de atender la llegada de Juan lo llamaremos encuesta.

Veamos ahora otro método. En esta ocasión simplemente se recuesta en el sofá de la sala y comienza a leer su libro, cuando Juan llegue debe tocar el timbre de la puerta y esperar unos momentos a que le atiendan. Cuando usted oye sonar el timbre, interrumpe la lectura, marca la página donde se quedó y va hasta la puerta para atender a la persona que toca el timbre. Una vez que Juan o la persona que ha tocado el timbre, se marcha, usted regresa a su asiento y retoma la lectura justo donde la dejó. Este último es un método más eficiente que el anterior porque le deja más tiempo para leer y elimina algunos inconvenientes como el de que Juan nunca llegue o se marche antes de que usted abra la puerta. Es, en principio, un método simple pero muy eficaz y eficiente, lo llamaremos atención por interrupción.

El primero de ellos, la encuesta, es un método eficaz, pero poco eficiente porque requiere realizar lecturas constantes y muchas veces innecesarias del estado del proceso que queremos atender. Sin embargo, es muy utilizado en la programación de microcontroladores porque resulta fácil de aprender, la implementación de código con este método es menos compleja y no requiere de hardware especial para llevarla adelante. Por otra parte, la encuesta, tiene muchas deficiencias que con frecuencia obligan al diseñador a moverse hacia otros horizontes

El mundo está lleno de situaciones; de las cuales no podemos determinar ni cuando, ni como ni por qué se producen, en la mayoría de los casos lo único que podemos hacer es enterarnos de que determinada situación, asociada a un proceso, ha ocurrido. Para ello seleccionamos alguna condición o grupo de condiciones que nos indican que el proceso que nos interesa debe ser atendido, a este fenómeno, en el cual se dan las condiciones que nos interesa conocer, lo llamaremos evento. En el segundo ejemplo vemos que para atender a Juan, este debe tocar el timbre, por tanto, la llegada de Juan es el proceso que debemos atender y el sonido del timbre es el evento que nos indica que Juan ha llegado.

El método de atención a procesos por interrupción, visto desde la óptica del ejemplo que utilicé para mostrarlo, es más simple que el de la encuesta, pero no es cierto, el método se complica porque requiere que el microprocesador incorpore circuitos adicionales para registrar los eventos que le indican que debe atender al proceso asociado y comprender estos circuitos y su dinámica no es una tarea sencilla.

Los circuitos para la atención a las interrupciones y todas las tareas que debe realizar el procesador para atender al proceso que lo interrumpe son bastante complejos y requieren una visión diferente de la que estamos acostumbrados a tener de nuestro mundo.

Los seres humanos no estamos conscientes de las interrupciones, en nuestro organismo existen mecanismos que nos interrumpen constantemente, para ello tenemos a nuestro sistema sensorial, pero no somos conscientes del proceso de interrupción, aunque sí de la atención a las interrupciones. Eso es porque incorporamos mecanismos que nos sacan rápidamente de la tarea que estemos haciendo para atender una situación que no puede o no debe esperar mucho tiempo. Bien, esa misma es la idea que se incorpora en los microprocesadores para atender procesos que no pueden esperar o que no sabemos cuando deben ser atendidos porque ello depende de determinadas condiciones.

La cosa se complica en la secuencia de acciones a realizar desde el momento en que se desencadena el proceso de interrupción, hasta que se ejecuta el programa que lo atiende, y en la secuencia de acciones posteriores a la atención. Piense en cuantas cosas debe hacer su organismo ante una interrupción, utilicemos el segundo ejemplo para atender la llegada de Juan. Piense en cuantas cosas su cerebro hace a espaldas de su conciencia, desde el momento en que suena el timbre hasta que usted se encuentra listo (consciente de que es probable que Juan ha llegado) para abrir la puerta, y todo lo que su cerebro debe trabajar para retomar la lectura después que Juan se ha marchado. Todo eso, excepto abrir la puerta y atender a Juan, lo hacemos de forma “inconsciente” porque para ello tenemos sistemas dedicados en nuestro organismo, pero en el mundo de los microcontroladores debemos conocer todos esos detalles para poder utilizar los mecanismos de interrupción.

Los procesos de atención a interrupciones tienen la ventaja de que se implementan por hardware ubicado en el procesador, así que es un método rápido de hacer que el procesador se dedique a ejecutar un programa especial para atender eventos que no pueden esperar por mecanismos lentos como el de encuesta.

En términos generales, un proceso de interrupción y su atención por parte del procesador, tiene la siguiente secuencia de acciones:

1. En el mundo real se produce el evento para el cual queremos que el procesador ejecute un programa especial, este proceso tiene la característica de que no puede esperar mucho tiempo antes de ser atendido o no sabemos en que momento debe ser atendido.
2. El circuito encargado de detectar la ocurrencia del evento se activa, y como consecuencia, activa la entrada de interrupción del procesador.
3. La unidad de control detecta que se ha producido una interrupción y “levanta” una bandera para registrar esta situación; de esta forma si las condiciones que

- provocaron el evento desaparecen y el circuito encargado de detectarlo desactiva la entrada de interrupción del procesador, esta se producirá de cualquier modo, porque ha sido registrada.
4. La unidad de ejecución termina con la instrucción en curso y justo antes de comenzar a ejecutar la siguiente comprueba que se ha registrado una interrupción
 5. Se desencadena un proceso que permite guardar el estado actual del programa en ejecución y saltar a una dirección especial de memoria de programas, donde está la primera instrucción de la subrutina de atención a interrupción.
 6. Se ejecuta el código de atención a interrupción, esta es la parte “consciente” de todo el proceso porque es donde se realizan las acciones propias de la atención a la interrupción y el programador juega su papel.
 7. Cuando en la subrutina de atención a interrupción se ejecuta la instrucción de retorno, se desencadena el proceso de restauración del procesador al estado en que estaba antes de la atención a la interrupción.

Como podemos observar, el mecanismo de interrupción es bastante complicado, sin embargo tiene dos ventajas que obligan a su implementación: la velocidad y su capacidad de ser asíncrono. Ambas de conjunto permiten que aprovechemos al máximo las capacidades de trabajo de nuestro procesador.

Los mecanismos de interrupción no solo se utilizan para atender eventos ligados a procesos que requieren atención inmediata sino que se utilizan además para atender eventos de procesos asíncronos.

Las interrupciones son tan eficaces que permiten que el procesador actúe como si estuviese haciendo varias cosas a la vez cuando en realidad se dedica a la misma rutina de siempre, ejecutar instrucciones una detrás de la otra.

Periféricos

Cuando observamos la organización básica de un microcontrolador, señalamos que dentro de este se ubican un conjunto de periféricos. A continuación describiremos algunos de los periféricos que con mayor frecuencia encontraremos en los microcontroladores.

Entradas y salidas de propósito general

También conocidos como puertos de E/S, generalmente agrupadas en puertos de 8 bits de longitud, permiten leer datos del exterior o escribir en ellos desde el interior del microcontrolador, el destino habitual es el trabajo con dispositivos simples como relés, LED, o cualquier otra cosa que se le ocurra al programador.

Algunos puertos de E/S tienen características especiales que le permiten manejar salidas con determinados requerimientos de corriente, o incorporan mecanismos especiales de interrupción para el procesador.

Típicamente cualquier pin de E/S puede ser considerada E/S de propósito general, pero como los microcontroladores no pueden tener infinitos pines, ni siquiera todos los pines que queramos, las E/S de propósito general comparten los pines con otros periféricos. Para usar un pin con cualquiera de las características a él asignadas debemos configurarlo mediante los registros destinados a ellos.

Temporizadores y contadores

Son circuitos sincrónicos para el conteo de los pulsos que llegan a su poder para conseguir la entrada de reloj. Si la fuente de un gran conteo es el oscilador interno del microcontrolador es común que no tengan un pin asociado, y en este caso trabajan como temporizadores. Por otra parte, cuando la fuente de conteo es externa, entonces tienen asociado un pin configurado como entrada, este es el modo contador.

Los temporizadores son uno de los periféricos más habituales en los microcontroladores y se utilizan para

muchas tareas, como por ejemplo, la medición de frecuencia, implementación de relojes, para el trabajo de conjunto con otros periféricos que requieren una base estable de tiempo entre otras funcionalidades. Es frecuente que un microcontrolador típico incorpore más de un temporizador/contador e incluso algunos tienen arreglos de contadores. Como veremos más adelante este periférico es un elemento casi imprescindible y es habitual que tengan asociada alguna interrupción. Los tamaños típicos de los registros de conteo son 8 y 16 bits, pudiendo encontrar dispositivos que solo tienen temporizadores de un tamaño o con más frecuencia con ambos tipos de registro de conteo.

Conversor analógico/digital

Como es muy frecuente el trabajo con señales analógicas, estas deben ser convertidas a digital y por ello muchos microcontroladores incorporan un [conversor analógico-digital](#), el cual se utiliza para tomar datos de varias entradas diferentes que se seleccionan mediante un multiplexor.

Las resoluciones más frecuentes son 8 y 10 bits, que son suficientes para aplicaciones sencillas. Para aplicaciones en control e instrumentación están disponibles resoluciones de 12bit, 16bit y 24bit.⁶ También es posible conectar un convertidor externo, en caso de necesidad

Puertos de comunicación

Puerto serie

Este periférico está presente en casi cualquier microcontrolador, normalmente en forma de [UART](#) (*Universal Asynchronous Receiver Transmitter*) o [USART](#) (*Universal Synchronous Asynchronous Receiver Transmitter*) dependiendo de si permiten o no el modo sincrónico de comunicación.

El destino común de este periférico es la comunicación con otro microcontrolador o con una PC y en la mayoría de los casos hay que agregar circuitos externos para completar la interfaz de comunicación. La forma más común de completar el puerto serie es para comunicarlo con una PC mediante la interfaz [EIA-232](#) (más conocida como [RS-232](#)), es por ello

que muchas personas se refieren a la UART o USART como puerto serie RS-232, pero esto constituye un error, puesto que este periférico se puede utilizar para interconectar dispositivos mediante otros estándares de comunicación. En aplicaciones industriales se utiliza preferiblemente [RS-485](#) por sus superior alcance en distancia, velocidad y resistencia al ruido.

SPI

Este tipo de periférico se utiliza para comunicar al microcontrolador con otros microcontroladores o con periféricos externos conectados a él, por medio de una interfaz muy sencilla. Hay solo un nodo controlador que permite iniciar cualquier transacción, lo cual es una desventaja en sistemas complejos, pero su sencillez permite el aislamiento galvánico de forma directa por medio de [optoacopladores](#).

I2C

Cumple las mismas funciones que el SPI, pero requiere menos señales de comunicación y cualquier nodo puede iniciar una transacción. Es muy utilizado para conectar las tarjetas gráficas de las computadoras personales con los monitores, para que estos últimos informen de sus prestaciones y permitir la autoconfiguración del sistema de vídeo.

USB

Los microcontroladores son los que han permitido la existencia de este sistema de comunicación. Es un sistema que trabaja por [polling](#) (monitorización) de un conjunto de periféricos inteligentes por parte de un amo, que es normalmente un computador personal. Cada modo inteligente está gobernado inevitablemente por un microcontrolador.

Ethernet

Es el sistema más extendido en el mundo para [redes de área local](#) cableadas. Los microcontroladores más poderosos de 32 bits se usan para implementar periféricos lo suficientemente poderosos como para que puedan ser

accedidos directamente por la red. Muchos de los [enrutadores](#) caseros de pequeñas empresas están contruidos sobre la base de un microcontrolador que hace del cerebro del sistema.

Can

Este protocolo es del tipo [CSMA/CD](#) con tolerancia a elevados niveles de tensión de modo común y orientado al tiempo real. Este protocolo es el estándar más importante en la industria automotriz ([OBD](#)). También se usa como capa física del "[field bus](#)" para el control industrial.

Otros puertos de comunicación

Hay una enorme cantidad de otros buses disponibles para la industria automotriz (linbus) o de medios audiovisuales como el i2s, [IEEE 1394](#). El usuario se los encontrará cuando trabaje en algún área especializada.

Comparadores

Son circuitos analógicos basados en amplificadores operacionales que tienen la característica de comparar dos señales analógicas y dar como salida los niveles lógicos '0' o '1' en dependencia del resultado de la comparación. Es un periférico muy útil para detectar cambios en señales de entrada de las que solamente nos interesa conocer cuando está en un rango determinado de tensión.

Modulador de ancho de pulsos

Los PWM ([Pulse Width Modulator](#)) son periféricos muy útiles sobre todo para el control de motores, sin embargo hay un grupo de aplicaciones que pueden realizarse con este periférico, dentro de las cuales podemos citar: inversión DC/AC para [UPS](#), conversión digital analógica D/A, control regulado de luz (dimming) entre otras.

Memoria de datos no volátil

Muchos microcontroladores han incorporado estos tipos de memoria como un periférico más, para el almacenamiento de datos de configuración o de los procesos que se controlan. Esta memoria es independiente de la memoria de datos tipo

RAM o la memoria de programas, en la que se almacena el código del programa a ejecutar por el procesador del microcontrolador.

Muchos de los microcontroladores PIC incluyen este tipo de memoria, típicamente en forma de memoria [EEPROM](#), incluso algunos de ellos permiten utilizar parte de la memoria de programas como memoria de datos no volátil, por lo que el procesador tiene la capacidad de escribir en la memoria de programas como si esta fuese un periférico más.

Familias de microcontroladores

Los microcontroladores más comunes en uso son:

Empresa	8 bits	16 bits	32 bits
Atmel	AVR (mega y tiny), 89Sxxxx familia similar 8051		SAM7 (ARM7TDMI), SAM3 (ARM Cortex-M3), SAM9 (ARM926), AVR32
Freescale (antes Motorola)	68HC05 , 68HC08 , 68HC11 , HCS08	68HC12 , 68HCS12 , 68HCSX12 , 68HC16	683xx , PowerPC , ColdFire
Holttek	HT8		
Intel	MCS-48 (familia 8048) MCS51 (familia 8051) 8xC251	MCS96 , MXS296	x



TECNOLÓGICO
NACIONAL DE MÉXICO®



TECNM
CAMPUS
CHIHUAHUA II

National Semiconductor	COP8	x	x
Microchip	Familia 10f2xx Familia 12Cxx Familia 12Fxx, 16Cxx y 16Fxx 18Cxx y 18Fxx	PIC24F, PIC24H y dsPIC30FXX, dsPIC33F con motor dsp integrado	PIC32
NXP Semiconductors (antes Philips)	80C51	XA	Cortex-M3, Cortex-M0, ARM7, ARM9
Renesas (antes Hitachi, Mitsubishi y NEC)	78K , H8	H8S , 78K0R , R8C , R32C/M32C/M16C	RX , V850 , SuperH , SH-Mobile , H8SX
STMicroelectronics	ST 62 , ST 7		STM32 (ARM7)
Texas Instruments	TMS370	MSP430	C2000, Cortex-M3 (ARM), TMS570 (ARM)
Zilog	Z8 , Z86E02		

Observación: Algunas arquitecturas de microcontrolador están disponibles por tal cantidad de vendedores y en tantas variedades, que podrían tener, con total corrección, su

propia categoría. Entre ellos encontramos, principalmente, las variantes de [Intel 8051](#) y [Z80](#).

Véase también

- [Lista de microcontroladores comunes](#)
- [Controladora de periférico](#)
- [Microprocesador](#)
- [Microbótica](#)
- [PIC16F87X](#)
- [Intel HEX](#)
- [Informática de sistemas](#)
- [Ingeniería Técnica en Informática de Gestión](#)
- [BASIC Stamp](#)
- [Arduino](#)

Referencias

1. [↑] Vienna University of Technology, Introduction To Microcontrollers (Gunther Gridling, Bettina Weiss), 26 de febrero de 2007
2. [↑] Augarten, Stan (1983). *The Most Widely Used Computer on a Chip: The TMS 1000*. State of the Art: A Photographic History of the Integrated Circuit. New Haven y New York: Ticknor & Fields. ISBN 0-89919-195-9. Consultado el 2 de marzo de 2012.
3. [↑] Saltar a:^a ^b https://web.archive.org/web/20120619154428/http://archive.computerhistory.org/resources/access/text/Oral_History/102658328.05.01.acc.pdf Oral History Panel on the Development and Promotion of the Intel 8048 Microcontroller, Computer History Museum oral history, 2008, consultado el 28 de junio de 2011, página 4
4. [↑] «Copia archivada». Archivado desde [el original](#) el 22 de agosto de 2016. Consultado el 3 de marzo de 2012.
5. [↑] "Atmel's Self-Programming Flash Microcontrollers" de Odd Jostein Svendsli 2003
6. [↑] [microcontroladores "analógicos"](#)

QUÉ ES UN MICROCONTROLADOR

Es un equipo con las mismas características de una computadora, solo que su tamaño es más pequeño. Tiene un CPU (Central Processing Unit) por sus siglas en inglés, una memoria RAM y una memoria ROM. Es el cerebro de un sistema informático y el motor que activa el funcionamiento de un equipo.

Su utilidad está presente en muchas áreas de la vida cotidiana, en la industria cumple una tarea fundamental, ya que es utilizado como complemento en la automatización de diversas operaciones.

¿Para qué sirve un microcontrolador?

Con un **microcontrolador** tenemos la posibilidad de realizar múltiples tareas, tales como la administración de entrada y salida en un proceso informático determinado. En el sector industrial es frecuente ver su aplicación en controladores y otros sistemas de automatización que detallaremos más adelante.

Por otro lado, en la mayoría de los dispositivos tecnológicos que usamos está presente el microcontrolador. Ellos hacen posible el funcionamiento de ordenadores, celulares, calculadoras, laptop, relojes, alarmas, entre otros equipos.

Tipos de microcontrolador

Los **microcontroladores** se clasifican de diferentes maneras, de acuerdo a los bits, a las instrucciones y el tipo de memoria.

De acuerdo a sus bits lo podemos encontrar de 8, 16 o 32 bits (unidad de información). Cada uno de estos controladores es utilizado según la capacidad de bits que amerite el sistema, por ejemplo, los sistemas automáticos requieren 32 bits.

Con relación al tipo de instrucción, los hay de indicaciones reducidas, los cuales se conocen con el código RISC-RISC y los de instrucciones difíciles. Estos pueden hacer varias tareas simples a la vez.



Los tipos según la memoria se clasifican en Harvard Memory Architecture Microcontroller y Microcontrolador de arquitectura de memoria Princeton

¿Cómo funciona un microcontrolador?

El **microcontrolador** está compuesto por un circuito integrado donde convergen varios elementos. Su funcionamiento dependerá del programa para el cual fue implementado. En la memoria del dispositivo se almacenan los datos, para que el CPU realice las instrucciones de dicho programa.

Es importante mencionar que antes de grabar información en un microcontrolador, es necesario agrupar los datos a formato hexadecimal para que el dispositivo pueda funcionar.

Características de los microcontroladores

- Son compatibles para reducir el consumo de energía eléctrica
- Estos dispositivos se pueden convertir en un unidad central de procesamiento con pocos circuitos integrados de manera externa
- Poseen una pequeña cantidad de memoria aleatoria
- Cuentan con elementos de entrada y salida como el temporizador y los buses de interfaz
- Disponen de ondas de alta frecuencia, las cuales configuran los impulsos del reloj que se usa para la regularización de todos los métodos donde se aplique

Microcontrolador en los procesos industriales

El auge que ha tenido la automatización en los procesos industriales es inminente, cada día se siguen estudiando posibilidades que permitan mantener una eficiente automatización de las operaciones sin generar costos excesivos.

El **microcontrolador** representa una opción para aquellas industrias que no tienen la capacidad de incorporar un PLC a sus operaciones.

Este dispositivo permite programar y controlar sistemas de manera eficiente garantizando resultados óptimos.



Estos equipos pueden ser usados en las industrias en actividades de vibración, temperatura, choque y longevidad. Los microcontroladores de uso industrial, tienen un rango de trabajo amplio, con el cual puede ajustarse a distintas actividades dentro de la empresa.

Partes de un microcontrolador

- CPU
- Memoria RAM
- Memoria ROM
- Líneas de entrada y salida

Ventajas de los microcontroladores

- Son de tamaño reducido
- Poseen gran flexibilidad
- Puede sustituir a otros circuitos integrados sin generar cambios en el sistema
- Sus salidas y entradas se pueden configurar de acuerdo a las necesidades de cada usuario
- Un microcontrolador puede ser reprogramado
- Con un microcontrolador se pueden programar, controlar y combinar distintas operaciones a la vez.

Fabricantes de microcontroladores

A continuación presentamos las marcas de **microcontroladores** reconocidas por la calidad de sus productos:

AMCC, Altera, ELAN Microelectronics, Freescale Semiconductor, EPSON Semiconductor, Dallas Semiconductor, Analog Devices, Cypress MicroSystems, Energy Micro AS, Charmed Labs, Intel, Atmel, Infineon, Fujitsu, Holtek, Xilinx, Toshiba, Xemics, ZiLOG, Western Design Center, Ubicom, Lattice Semiconductor, National Semiconductor, Rabbit Semiconductor, Silicon Motion, Renesas Technology, Rabbit Semiconductor, NXP Semiconductors, NEC, Parallax, Silabs, STMicroelectronics, Texas Instruments



Mejores controladores para ser aplicados en procesos industriales

Estos son los controladores de uso industrial de acuerdo a su calidad, costo y adaptación a las distintas operaciones

- Digital Discovery: analizador lógico portátil
- KeeYees 2 Piezas Descargador Programador para USBASP para ISP con Cable y Placa Adaptadora de 10 Pines a 6 Pines
- SODIAL(R) Mini Nano V3.0 ATmega328P minicontrolador tablero w-USB Cable para Arduino
- AZDelivery ESP8266 ESP-12F versión mejorada de ESP-12E- módulo inalámbrico remoto WLAN WIFI para Arduino-
- ARCELI PIC ICD2 PICKIT 2 Programador PICKIT 3 PICKIT2 PICKIT3 Programador de Asientos Centrales
- HiLetgo Mega2560 R3 ATMEGA16U2 CH340 Microcontroller Board Development Board Compatible to Arduino With USB Cable
- C-NSC DIP-18 LM3914N-1 LM3914N-1 LM3914N-1 controladores de pantalla LED
- ARCELI ST-Link - V2 (Versión CN) ST MCU Microcontrolador STM8 STM32 JTAG SWD

Importancia de los microcontroladores en la vida cotidiana

Los **microcontroladores** hoy día están presentes en distintos espacios, lo podemos ver en artefactos del hogar como microondas, televisores, equipos de sonido y computadoras, pero también nos beneficiamos de este instrumento de la ingeniería electrónica cuando esperamos el cambio de la luz del semáforo.

En el campo industrial, su utilidad no puede pasar desapercibida, ya que sus beneficios forman parte de la calidad de muchos de los procesos de automatización que se desarrollan en las empresas, permitiendo tener resultados favorables para el fabricante como para el consumidor final.

En conclusión, las ciudades inteligentes de estos tiempos tienen un integrante en común; su nombre es microcontrolador.

¿Cuál es la diferencia entre microprocesadores y microcontroladores?

Los microprocesadores y microcontroladores son los componentes internos de los dispositivos electrónicos. Un microprocesador es una unidad de procesamiento muy pequeña dentro de una CPU. Es un circuito integrado único en un chip de computadora que realiza diversas funciones aritméticas y lógicas en señales digitales. Varias docenas de microprocesadores trabajan juntos dentro de servidores de alto rendimiento para el procesamiento y el análisis de datos.

Por otro lado, un microcontrolador es la unidad de computación básica dentro de los dispositivos electrónicos inteligentes, como las lavadoras y los termostatos. Es una computadora muy pequeña con sus propios sistemas de RAM, ROM y E/S, todos integrados en un solo chip. Puede procesar señales digitales y responder a las entradas del usuario, pero su capacidad de computación es limitada.

¿Cuáles son las similitudes entre los microprocesadores y los microcontroladores?

Los microprocesadores y microcontroladores son chips de computadora centralizados que proporcionan inteligencia a computadoras personales y dispositivos electrónicos. Están contruidos con circuitos integrados de semiconductores y comparten ciertas partes internas.

Circuito integrado

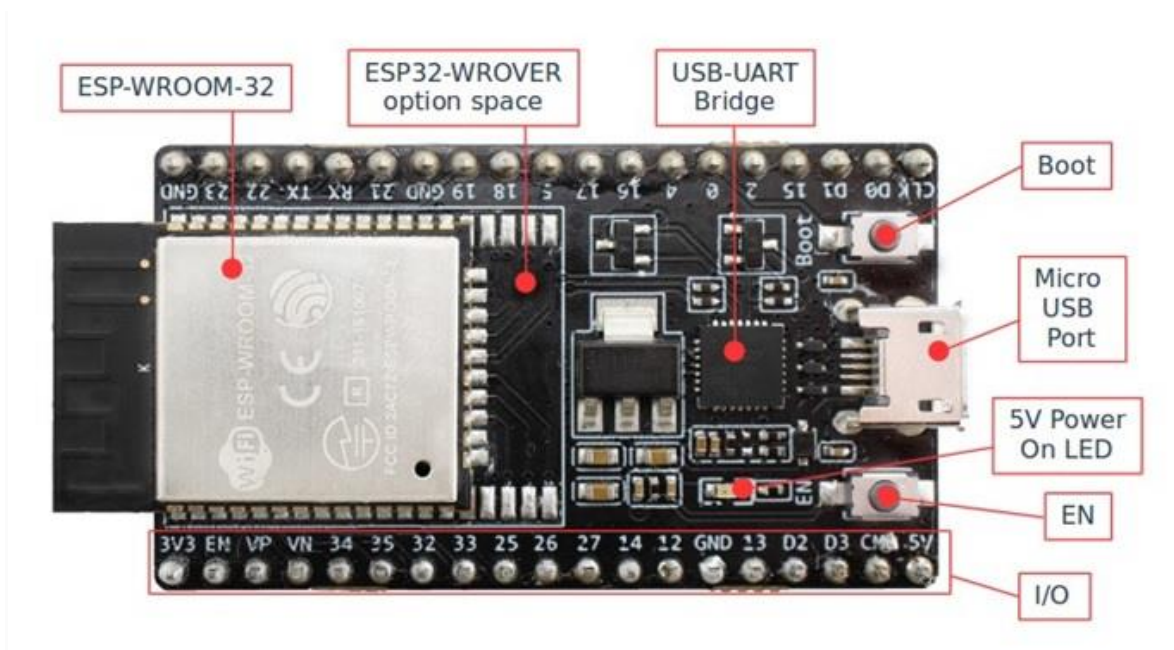
Tanto los microprocesadores como los microcontroladores son componentes semiconductores contruidos en un circuito integrado. Un circuito integrado es un chip cuadrado o rectangular muy pequeño que contiene miles o incluso millones de componentes electrónicos. Los circuitos integrados permiten a los ingenieros reducir el tamaño de los circuitos electrónicos.

CPU

Tanto los microprocesadores como los microcontroladores tienen una CPU. Una CPU es la parte centralizada del chip de la computadora que procesa las instrucciones proporcionadas por las aplicaciones o el firmware. La CPU también tiene un módulo especial de unidad aritmética lógica (ALU). Una ALU calcula valores matemáticos y evalúa los problemas lógicos basándose en las instrucciones de la computadora.

Registros

Los registros son módulos de memoria que la CPU utiliza para el procesamiento. La CPU almacena de manera temporal instrucciones o datos binarios antes, durante y después de procesarlos. Tanto los microprocesadores como los microcontroladores se construyen con registros internos, aunque los microcontroladores suelen tener más registros que los microprocesadores.



Diferencias arquitectónicas: microprocesadores frente a microcontroladores

A pesar de adoptar la forma de chips de computadora, los microprocesadores y microcontroladores se construyen con diferentes arquitecturas.

Los microprocesadores están diseñados con la arquitectura Von Neumann, en la que un programa y los datos residen en el mismo módulo de memoria. Mientras tanto, los microcontroladores utilizan la arquitectura Harvard, que separa la memoria del programa del espacio de datos.

Los microprocesadores tienen más componentes de circuitos integrados que los microcontroladores. Esta diferencia arquitectónica afecta a las consideraciones de diseño de los microprocesadores y microcontroladores en aplicaciones informáticas y de sistemas integrados.

[Más información sobre los chips de computadora »](#)

Memoria

Los microprocesadores no tienen módulos de memoria interna para almacenar los datos de las aplicaciones. Los ingenieros deben conectar el microprocesador a los almacenamientos de memoria externos, como la ROM y la RAM, con un bus externo.

Un bus es un conjunto de conexiones eléctricas paralelas que permiten al microprocesador enviar y recibir datos de otros dispositivos. Hay tres tipos de buses:

- Un bus de datos transmite datos
- Un bus de direcciones transmite información sobre dónde almacenar y recuperar datos
- Un bus de control transmite señales para coordinarlas con otros componentes eléctricos

Los tres funcionan de manera colectiva en un sistema de microprocesador.

Por otro lado, los microcontroladores están contruidos con memorias ROM y RAM internas. Un microcontrolador utiliza un bus interno para interactuar con los módulos de memoria integrados.

Periféricos

Los periféricos son temporizadores, comunicaciones, E/S y otras capacidades que permiten a los microcontroladores o microprocesadores interactuar con componentes o usuarios externos.

El microprocesador no tiene periféricos integrados en su circuito integrado. En cambio, los periféricos se conectan de manera externa para ampliar los casos de uso del microprocesador más allá del procesamiento matemático y lógico.

Por el contrario, los microcontroladores se conectan a los periféricos del chip mediante un bus de control interno. Esto permite que el microcontrolador controle los dispositivos electrónicos con un mínimo de piezas adicionales o incluso sin ellas.

Capacidad computacional

Los microprocesadores son potentes chips de computadora capaces de realizar tareas matemáticas y computacionales complejas. Por ejemplo, puede ejecutar un software de procesamiento estadístico porque el microprocesador admite el funcionamiento en punto flotante.

Por el contrario, los microcontroladores tienen una potencia de procesamiento comparativamente menor y rara vez admiten el cálculo de punto flotante. En cambio, se centran en implementar una lógica específica, como controlar la temperatura de un calentador en función de varios sensores.

Otras diferencias clave: microprocesadores frente a microcontroladores

Los microprocesadores permiten operaciones de computación versátiles en computadoras personales y servidores empresariales. Mientras tanto, los microcontroladores permiten que los sistemas integrados analicen y respondan a las entradas en tiempo real.

Cuando los ingenieros desarrollan sistemas con microprocesadores y microcontroladores, tienen en cuenta estas diferencias.

Velocidad del reloj

Los microprocesadores proporcionan capacidades de computación sólidas y de alta velocidad para diversas aplicaciones. Un procesador de computadora moderno funciona en el rango de gigahercios (GHz). Esto permite que un sistema de computación realice cálculos matemáticos complejos y devuelva los resultados con rapidez.

Si bien la velocidad del microcontrolador ha aumentado a lo largo de las décadas, es mucho menor que la velocidad de procesamiento del microprocesador. Según su propósito, la velocidad del reloj de un microcontrolador oscila entre kilohercios (kHz) y cientos de megahercios (MHz). A pesar del rango de velocidad inferior, un microcontrolador puede funcionar de manera óptima dentro de su ámbito de aplicación dedicado.

Tamaño del circuito

Un microprocesador no puede funcionar por sí solo. Se basa en partes externas, como chips de comunicación, puertos de E/S, RAM y ROM, para formar un sistema de computación completo. Por lo tanto, un circuito basado en un microprocesador consiste en un bus de direcciones y datos que conecta muchos periféricos y chips de memoria. Incluso con los avances en las tecnologías de placas de circuito impreso (PCB), un sistema de microprocesador requiere un espacio considerable.

Sin embargo, el microcontrolador proporciona un diseño que ahorra espacio con un circuito más simple. La mayoría de los componentes adicionales que necesita un sistema basado en microprocesadores están disponibles fácilmente en el mismo chip. En lugar de utilizar componentes separados de manera individual, los ingenieros utilizan un solo microcontrolador al diseñar dispositivos electrónicos. Esto permite disponer de más espacio en la placa de circuito electrónico, lo que permite a los ingenieros producir sistemas compactos.

Consumo de energía

Los microprocesadores suelen funcionar a una velocidad mayor que los microcontroladores y consumen más energía, por lo que requieren una fuente de alimentación externa. Del mismo modo, un sistema de computación basado en una unidad de microprocesador tiene un mayor consumo total de energía debido a la gran cantidad de componentes adicionales.

Mientras tanto, los microcontroladores están diseñados para funcionar de manera eficiente con una potencia mínima. Además, la mayoría de los microcontroladores cuentan con características de ahorro de energía, de las que carecen los microprocesadores.

Por ejemplo, un microcontrolador puede activar el modo de ahorro de energía y consumir energía limitada cuando no procesa datos. Los microcontroladores también pueden apagar los periféricos internos que no estén en uso para ahorrar energía. Esto hace que los microcontroladores sean ideales para crear una aplicación dedicada de bajo consumo que funcione con energía almacenada.

Sistema operativo

En aplicaciones prácticas, los microprocesadores requieren un sistema operativo que proporcione las funcionalidades adecuadas. Sin un sistema operativo, los



usuarios tendrían que instruir al microprocesador en lenguaje ensamblador o binario.

Mientras tanto, los microcontroladores no requieren un sistema operativo para funcionar. Sin embargo, existen sistemas operativos específicos que ayudan a los microcontroladores de gama media y alta a funcionar de manera más eficiente.

Conectividad

Los microprocesadores manejan tecnologías de comunicación más diversas que los microcontroladores. Por ejemplo, un microprocesador procesa datos USB 3.0 o Gigabit Ethernet de alta velocidad sin un procesador secundario.

Sin embargo, la mayoría de los microcontroladores necesitan un procesador especial para la conectividad de datos de alta velocidad.

Costo

Un circuito integrado de microprocesador solo consta de la CPU, la unidad aritmética lógica (ALU) y los registros, lo que reduce el costo de fabricación por unidad. Mientras tanto, un solo microcontrolador tiene una arquitectura interna más compleja y, por lo general, es más caro que un microprocesador.

Sin embargo, un sistema basado en microprocesadores es más caro, ya que requiere componentes adicionales. Por el contrario, un microcontrolador es autosuficiente para la aplicación elegida.

El microcontrolador requiere menos componentes adicionales, lo que se traduce en sistemas basados en microcontroladores más baratos. Por ejemplo, la placa de circuito de un acondicionador de aire con microcontrolador cuesta menos que la placa base de un ordenador con microprocesadores.

Casos de uso: microprocesadores frente a microcontroladores

Tanto los microprocesadores como los microcontroladores son componentes electrónicos útiles cuando se aplican a los casos de uso adecuados.

Utilice un microprocesador si necesita una potencia de procesamiento sólida para tareas de computación complejas o impredecibles. Los microprocesadores se utilizan en todo tipo de dispositivos de computación, como servidores, computadoras de sobremesa y dispositivos de computación móviles. Las

organizaciones utilizan servidores con muchos microprocesadores para la computación de alto rendimiento y para ejecutar aplicaciones de inteligencia artificial (IA).

Por otro lado, el microcontrolador es la mejor opción si está creando un sistema de control con un alcance definido de forma limitada. Los microcontroladores también son útiles para sistemas que requieren un bajo consumo de energía. Algunos microcontroladores pueden funcionar durante meses con solo una batería pequeña. Por ejemplo, un sistema doméstico inteligente funciona con microcontroladores. Los dispositivos compactos como los drones o los reproductores de audio portátiles también contienen microcontroladores.

Resumen de las diferencias: microprocesadores frente a microcontroladores

	Microprocesador	Microcontrolador
Memoria	Requiere memoria externa y almacenamiento de datos.	Módulos de chips de memoria (ROM, RAM).
Periféricos	Necesita piezas adicionales. Se conecta con el bus externo.	Periféricos integrados en el chip (temporizadores, puertos de E/S, conversor de señal).
Capacidad computacional	Capaz de realizar tareas de computación complejas.	Limitado a la lógica de la aplicación específica.
Velocidad del reloj	Muy rápido. Rango de GHz.	Rápido pero más lento que los microprocesadores. Rango de kHz a MHz.
Consumo de energía	Alto consumo de energía. Sin modo de ahorro de energía.	Consume una cantidad mínima de energía. Modos de ahorro de energía integrados.
Sistema operativo	Requiere sistemas operativos.	El sistema operativo es opcional para algunos microcontroladores.
Conectividad	Gestiona la transferencia de datos a alta velocidad. Soporta USB 3.0 y Gigabit Ethernet.	Soporta comunicaciones de baja a moderada velocidad. Interfaz periférica serial (SPI) e I ² C. Receptor-transmisor asíncrono universal (UART).
Costo	Caro debido a los componentes adicionales.	Más económico porque un solo circuito integrado proporciona múltiples funcionalidades.
Caso de uso	Para computación genérica o sistemas que requieren una	Para sistemas compactos, dispositivos alimentados por baterías o de



	capacidad sólida.	computacional	procesamiento lógico.
--	----------------------	---------------	-----------------------

¿Cómo puede ayudarle AWS con sus necesidades de desarrollo de microprocesadores y microcontroladores?

Amazon Web Services (AWS) satisface sus necesidades de desarrollo de microcontroladores y microprocesadores con los recursos y la infraestructura pertinentes.

Puede usar FreeRTOS para crear aplicaciones de microcontroladores modulares que se conecten con la nube. FreeRTOS es un sistema operativo de código abierto, con neutralidad de la nube y de tiempo real que ofrece un kernel rápido, fiable y receptivo. AWS proporciona bibliotecas útiles con FreeRTOS, lo que le permite integrar las capacidades de Internet de las cosas (IoT) en el firmware del microcontrolador con mayor facilidad.

Amazon Elastic Compute Cloud (EC2) permite a las organizaciones implementar aplicaciones basadas en microprocesadores en la nube. Puede escalar el entorno de computación, o la instancia, de acuerdo con las especificaciones de la aplicación y la demanda continua. Ofrecemos diferentes tipos de instancias, incluidas las que utilizan procesadores ARM, Intel y AMD, para casi cualquier carga de trabajo.

Para comenzar con el desarrollo de microprocesadores y microcontroladores en AWS cree una cuenta hoy mismo.

La importancia y aplicaciones de los microcontroladores en la Electrónica

El campo de la Ingeniería Electrónica es fundamental en los avances tecnológicos, cuyo desarrollo y evolución se ha llevado a cabo de forma acelerada en las últimas décadas. Uno de los objetivos principales de la Electrónica es el resolver tareas diversas, siendo los microcontroladores uno de los dispositivos con más potencial para la solución de dichas problemáticas.

Los microcontroladores son pequeños chips o dispositivos que pueden ser programados para realizar acciones o instrucciones que nosotros deseemos. Son de bajo costo, prácticos y poderosos para circuitos que necesitan ahorrar espacio físico. Además, son utilizados para mejorar la fiabilidad del funcionamiento y disminuir el consumo en los circuitos.



Para comprender mejor el impacto de los microcontroladores se mencionan a continuación algunas aplicaciones:

Las **cerraduras electrónicas** tienen un funcionamiento basado en el control de múltiples variables, para determinar si se abre o se mantienen cerrada una



cerradura. Mediante la implementación de microcontroladores se pueden mejorar los protocolos de seguridad en estos sistemas, los cuales podrían estar basados en conectividad “*bluetooth*” codificados a un módulo único. Dicho módulo podría ser una llave con una clave o incluso se podría activar la apertura mediante el uso de celulares propios vía “*bluetooth*”, por cercanía a la cerradura. Este conjunto de tecnologías permite desarrollar sistemas de cierre o apertura configurados no únicamente por biométrica física, sino mediante codificaciones de voz con ayuda de las inteligencias artificiales comerciales.

Uno de los modelos más recientes permite la comunicación con Inteligencia Artificial en casa, por ejemplo, para solicitar la apertura y cierre de puertas, para hacer llamadas de emergencia en caso de peligro, etc.

Debido a la gran cantidad de instrucciones que cualquier microcontrolador puede almacenar, son considerados una excelente opción en aplicaciones de la **robótica móvil**, tanto para la resolución de tareas específicas como para tareas generales. Otras de las ventajas del uso de los microcontroladores es que se pueden utilizar diferentes lenguajes de programación (C, C++ y Ensamblador), tienen diferentes puertos para la programación de diversas funciones y aplicaciones, así como el beneficio de que sus instrucciones no son estrictamente para resolver un único problema.

Otra aplicación de los microcontroladores en la robótica es su uso en objetos pequeños, llamados **mini robots**. Esto es posible, ya que los microcontroladores pueden ser controlados remotamente mediante una computadora, la cual procesa imágenes enviadas por éste, con la ventaja de que se pueden monitorear varios sensores al mismo tiempo. De la misma forma, se puede controlar la velocidad y dirección de avance. Un ejemplo de la aplicación de microcontroladores en mini robots podría ser su uso en caso de sismos, ya que con estos se podrían revisar edificios colapsados en busca de personas atrapadas, con mayor accesibilidad y sin poner en mayor riesgo al personal de búsqueda. Además, se puede utilizar IA y comportamiento subconsciente, que puede ser aplicado en un robot que busque imitar el comportamiento de un animal, con fines específicos. Esto combinaría las funciones anteriores (sensores y comportamiento autónomo) con otras funciones que el microcontrolador tiene para la robótica móvil.

Los microcontroladores han aumentado su importancia en la vida de todos a lo largo de la historia. Esto ha ocurrido principalmente en el campo de la Ingeniería Electrónica y su implementación en casas para convertirlas en hogares inteligentes, aplicación que se conoce como **domótica** o **diseño de casas inteligentes**. Un microcontrolador podría servir como el cerebro de nuestro hogar, llevando a cabo



instrucciones como, por ejemplo: cambios de luz, aumento de la temperatura, cambios a través de sensores de movimiento o proximidad, etc.

El uso de microcontroladores en **vehículos automatizados** es indispensable, ya que estos reciben un gran volumen de información de múltiples sensores, sobre todo de aquellos que están en niveles de automatización más altos y requieren funciones de comunicación de alta velocidad. Por lo tanto, es necesario un procesamiento de información más potente y acciones inmediatas ante amenazas de ciberseguridad “vehicle hacking”. Para cubrir estos retos se utilizan microcontroladores, los cuales poseen la ventaja de ser compactos y tienen la capacidad de conectar diversos dispositivos como sensores, cámaras y radares con el software para su control.

Los ejemplos antes mencionados, son solo unas de las tantas aplicaciones fascinantes que tienen los microcontroladores, situando a la Electrónica como una rama de la Ingeniería de mayor interés, utilidad e importancia para la solución de tareas y el logro de objetivos.

Escrito

Dra. Zizilia Zamudio Beltrán

por:

Con colaboración de los alumnos:

Daniela López Gutiérrez, Ranferi Molina Pérez, Daniel Ernesto Morales Palma y Fernanda Jacqueline Osorio Oliveros

Referencias

[1] Microcontrolador – qué es y para que sirve – HETPRO/TUTORIALES (2017). <https://hetpro-store.com/TUTORIALES/microcontrolador/#:~:>

[2] Turmero, P. (2020). Microcontroladores en robótica móvil – Monografias.com. <https://www.monografias.com/trabajos102/microcontroladores-robotica-mvil/microcontroladores-robotica-mvil.shtml>

[3] Ferrer N., Parladé X., José J., (2003), MICROCONTROLADORES EN ROBÓTICA MÓVIL, Tesis Ingeniería Técnica en Electrónica Industrial, Universidad Politécnica de Catalunya.



[4] Los microcontroladores de hoy en día. (2012). <https://microcontroladoresesv.wordpress.com/los-microcontroladores-de-hoy-en-dia/>

[5] Cabrera, A., (2015), La Domótica en México y su desarrollo en un microcontrolador. <https://www.testtechnology.net/single-post/2015/07/08/La-dom%C3%B3tica-en-M%C3%A9xico-y-su-desarrollo-en-un-microcontrolador>

[6] Aplicaciones de la Domótica (Usos, Funciones, Utilidades) – IECOR. (2020). <https://www.iecor.com/aplicaciones-de-la-domotica/>

[7] Microcontroladores para vehículos autónomos – diarioelectronicohoy.com. (2016). <https://www.diarioelectronicohoy.com/microcontroladores-vehiculos-autonomos/>

[8] Hyatt, K., Paukert, C., (2018), Los vehículos autónomos: entendiendo los distintos niveles de automatización, CNET en español. <https://www.cnet.com/es/noticias/vehiculos-autonomos-niveles-de-automatizacion/>

[9] Vehículos Automatizados para la Seguridad. (2018). <https://www.nhtsa.gov/es/tecnologia-e-innovacion/vehiculos-automatizados-para-la-seguridad>

Tags: 5G, Electrónica, IA, microcontroladores, robótica