

Procesador vs. Controlador (septiembre 2021)

Alejandro Garnung Menéndez

I. INTRODUCCIÓN

EN este documento pretenden recogerse, de manera concisa y concluyente, *todas* las diferencias que existen entre dos elementos fundamentales de la electrónica y la computación, que son los microprocesadores y los microcontroladores. El título de este informe suprime los prefijos micro- (del griego *mikrós* [pequeño], relacionado típicamente con dispositivos de escalas inferiores o iguales a micrómetros) de ambos términos, para hacer énfasis en que son las funciones de tales componentes lo que suele confundir a los iniciados en estos campos de estudio, a la hora de diferenciarlos. Después de valorar algunos importantes aspectos históricos y de resumir, muy concisamente, la definición de los dos, se citan y aclaran sus diferencias, para concluir estableciendo un marco general que introduce una idea de las aplicaciones básicas y comunes de estos dispositivos, así como induce una comprensión de su estructura interna y su aprovechamiento.

II. BREVE RESEÑA HISTÓRICA

Es razonable establecer el punto de partida del desarrollo de la tecnología, pertinente a estos dos dispositivos clave, en el comienzo de los circuitos integrados de las familias MOS y CMOS, en la década de 1960. La escalabilidad incesable de los transistores con tecnología de puerta de silicio (no bipolares) derivó, en 1971, a la fabricación, en un solo CI (chip), del microprocesador Intel 4004 (gracias a Federico Faggin [que grabó, originalmente, su nombre en el diseño del circuito] y otros desarrolladores de Intel y Busicom). Este antiguo y revolucionario dispositivo contaba con un reloj oscilador de 740 kHz, 4 bits de tamaño de instrucción (tamaño de palabra) (y almacenaba los datos en “paquetes” de 1 *nibble*), unos 2300 transistores, encapsulado DIP (*Dual In-line Package*) y arquitectura Harvard (sus instrucciones se almacenaban en la memoria ROM y los datos en otros registros de memoria separados [almacenamiento separado de programa y datos]). Más aún, por aquel entonces, Intel había decidido que toda esa la familia (4000) de transistores fuera encapsulada con solo 16 pines, por lo que este μ P (microprocesador) usaba un bus de 4 bits para multiplexar direcciones, instrucciones y datos. Es interesante comentar que este dispositivo es el que, generalmente, se considera como iniciador de la Ley de Moore. También cabe destacar que existen distintas opiniones sobre la autoría del primer microprocesador comercial, algunos la atribuyen a Lee Boysel (con su AL1 en 1969) o a Ray Holt (con su MP944), entre otros. Le siguió, en esa década, la serie (también de Intel) del 8080, 8085, 8086 y 8088 (de 8 bits) (en este último se basó la famosa IBM PC de 1981), la familia PPS-25 de

Fairchild Semiconductor, el 6501 y 6502 de Mostek o las series μ COM de NEC Corporation, entre otros.

El término “microprocesador” fue acuñado inicialmente por Viatron Computer Systems sobre 1970 (aunque usado para designar a sus máquinas computarizadas “System 21”) y se relacionó, desde el principio, con el concepto de “microprogramación” (Maurice Wilkes, 1951), con el que se describe la técnica de ejecutar las instrucciones de un programa a través de secuencias de instrucciones más elementales y se relaciona con controlar la CPU de un computador, rápidamente y mediante un programa especializado, en la ROM.

En el año 1971, Gary Moore y Michael Cochran (de Texas Instruments [TI]) crearon, presumiblemente, el primer microcontrolador, el TMS1802NC, rediseñado poco después dentro de la familia TM0100. Contaba con aprox. 5000 transistores, 3520 bits de ROM y 182 bits de memoria de acceso serial (SAM), aparte de los demás bloques y componentes internos que le otorgaban un amplio rango de funciones para su programación. Ideado para integrarse en una calculadora de cuatro funciones, casi solo necesitaba un *display* con sus drivers y un teclado como elementos extrínsecos.

En 1974, Texas Instruments introdujo al mercado la familia de microcontroladores TMS1000, descrita como un conjunto de “microcomputadores” PMOS de 4 bits con RAM, ROM, reloj interno y una ALU (Unidad Aritmética Lógica) integradas en un solo chip de s-c (semiconductor) (además de las líneas [y su control] de entrada y salida específicas). Este componente recibió esa denominación porque “recordaba” a una computadora en miniatura (su dependencia de periféricos extrínsecos era más baja) y proporcionaba una funcionalidad adicional a los (micro)procesadores, que era la flexibilidad, asegurando una eficiencia para aplicaciones específicas como la automoción, sistemas de sensores remotos u otros de control.

Intel lanzó sus primeros μ C (8048, 8035, 8748...) con la serie MSC-48, en 1976. 4 años más tarde se comercializó el popular 8051 (serie MSC-51). Si bien estos dispositivos, cada poco tiempo, solían presentar aún más memoria y eficiencia, menor costo, conjuntos de instrucciones más completos y software de desarrollo, no fue hasta principios de los 90 cuando la novedosa tecnología EEPROM (ROM programable y borrable eléctricamente) por fin apareció, permitiendo que los microcontroladores (y demás dispositivos) dependieran únicamente de señales eléctricas para (re)programarse. Antes de esta, las alternativas se basaban (entre otras) en otros dos métodos distintos. El primer tipo lo forman las memorias EPROM (ROM programable y borrable), que requerían de una “ventana” de cuarzo transparente intercalada entre un haz de luz ultravioleta y la propia memoria ROM. El segundo tipo era la memoria PROM, la cual no se podía programar más de una

vez (porque la ROM no era borrable).

Cabe mencionar que otra memoria, denominada “flash”, puede ser semejable a la memoria EEPROM; ambas son no volátiles pero, mientras la primera puede operar sobre varias posiciones de memoria durante una misma consigna, la segunda solo permite actuar sobre celdas de memoria individuales de cada vez.

El desarrollo inicial de la tecnología de la microelectrónica, en esta dirección, durante los 70 (lógica PMOS, arquitecturas de 4 bits, 8 bits, NMOS, 16 bits...) fue muy extenso, aun similar al de las décadas siguientes: la de los 80 (tecnología CMOS, 16 bits, 32 bits, aumento drástico de la escalabilidad de transistores [más de un millón], unidad de punto flotante integrada en el CI...), los 90 (velocidades de reloj mucho más altas, 64 bits...), los 2000 (computación paralela mediante procesos multinúcleo y multihilo, *overclocking*, refinamiento de sistemas de enfriamiento, más de mil millones de transistores integrados...), etc.

III. DEFINICIONES Y DIFERENCIAS

A mediados de los 40, John von Neumann (mientras ayudaba al desarrollo de una de las primeras computadoras [binarias] electrónicas, la EDVAC [Computadora Automática Variable Discreta Electrónica]) describió una arquitectura u organización interna de las computadoras, en la que su información binaria, de datos y del programa, se almacenaba en la misma dirección de memoria.

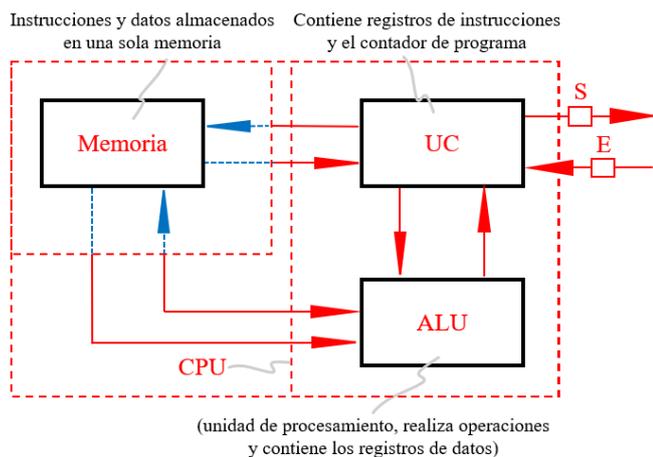


Fig. 1. Arquitectura típica de una máquina de von Neumann; lo que sería la Unidad Central de Procesamiento (CPU) estaría conformada por una Unidad de Control (UC) y una Unidad de Procesamiento (ALU...), una memoria que almacena datos e instrucciones y periféricos de entrada y salida. La acción simultánea de operaciones e instrucciones no es aplicable en este tipo de máquinas. Esto es el efecto de que usen un único bus común; se conoce como “cuello de botella de von Neumann” (limita la velocidad y el rendimiento del sistema).

Realmente, la conjunción de la UC y la ALU en un solo integrado es lo que caracteriza a un sistema para ser de tipo “microprocesador”. La adición de una memoria intrínseca es lo que lo caracteriza para ser un “microcomputador”.

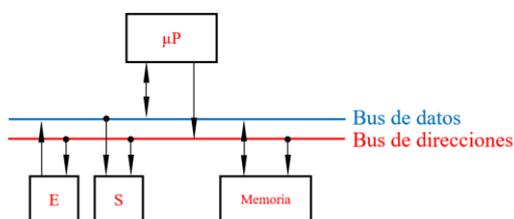


Fig. 2. Ejemplo de cómo se interconectaría un microprocesador (μP) a una serie de periféricos y componentes extrínsecos, tales como una memoria. La información (“tantos” bits de ancho) fluye a través de varios buses: un bus

de datos bidireccional y un bus de direcciones unidireccional (se omite en la figura un tercer bus, el de control, que se ocupa de decisiones de lectura, escritura, interrupciones, reloj ..., desde y hacia el microprocesador).

La arquitectura de conexión de la anterior figura se denomina “tipo Harvard”, basado en la computadora electromecánica Harvard Mark-I, de IBM (1944), en la que se almacenaban las instrucciones en cintas perforadas y los datos se manejaban con relés; esta filosofía de funcionamiento se conserva actualmente, mantiene alejados los datos y las instrucciones dentro de la memoria (al contrario que la arquitectura von Neumann, en la que el flujo de información y su acceso se realizaba a través de un solo bus). Cabe destacar que existe otro tipo de arquitectura derivada, la “Harvard modificada”, que flexibiliza la separación en memoria del tipo original (reserva cierta memoria de programa para datos, soporta cargas e inicializaciones de programas...).

Un microprocesador (MPU), en definitiva, es básicamente una unidad central de procesamiento (CPU, como la de un computador) integrada en un único chip de silicio. Profundizando aún más dentro de un microprocesador, se puede distinguir un tipo de memoria particular, la “memoria caché” es un tipo de memoria de rápido acceso y más “pequeña” que aquella memoria DRAM para la que puede trabajar como buffer independiente. El acceso aleatorio (memoria volátil; la información se borra si la memoria no está alimentada) puede entenderse como una manera no secuencial para acceder a los recursos en memoria, de manera que se tarda el mismo tiempo sin importar a qué porción de memoria se acceda (no hace falta leer la información anterior de cada vez); si bien la memoria DRAM (dinámica) necesita ser eventualmente reenergizada para no perder su contenido, la SRAM (memorias caché) no lo requiere, aunque al ser menos densa y más cara, su capacidad de almacenamiento se ve mermada, como se comentó antes. Una memoria RAM en un MPU de un PC (en su CPU) “debería” tener más memoria tipo caché que un microcontrolador, en condiciones similares.

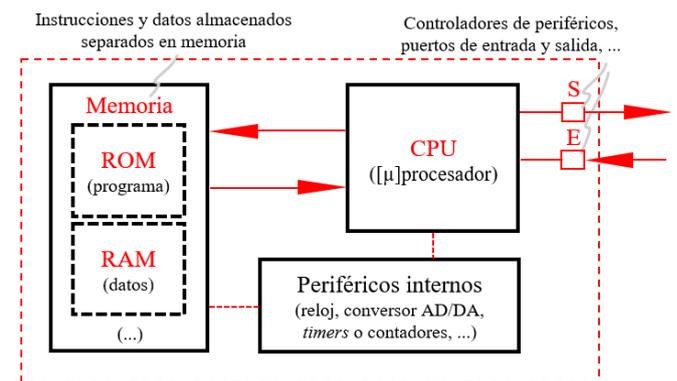


Fig. 3. Esquema ejemplificador de la estructura interna por bloques de un dispositivo microcontrolador (bloque discontinuo rojo). Sus buses de control internos se manejan basándose en la arquitectura Harvard (típicamente), antes comentada.

Resulta interesante comentar que una de las clasificaciones que se pueden hacer de los microprocesadores se elige en función del tipo de sus conjuntos de instrucciones (además de por su longitud de palabras, tamaño de registros...): pueden ser CISC (*Complex Instruction Set Computing*), si tienen un conjunto de instrucciones más completas y lentas (permiten realizar múltiples operaciones a bajo nivel en una sola instrucción [se agrupan instrucciones]), o RISC (*Reduced Instruction Set Computing*), si utiliza un número reducido de

instrucciones de carácter general y optimizadas (funcionando a mayor velocidad que las del otro, pero almacenando más [ocupando más espacio]).

Manteniendo, sobre todo, las ideas sacadas de las anteriores figuras presentadas, se pueden sintetizar, una por una, algunas de las diferencias más importantes entre un microcontrolador y un microprocesador.

A. Diferencias intuitivas

Pueden inducirse varias apreciaciones cualitativas que caracterizan a los dos dispositivos, conociendo con qué facilidad pueden ser capaces de manejar datos internos y externos, comunicarse con periféricos de salida y en términos de la aplicación en que se puedan implementar.

Un claro ejemplo es el caso de los MPU integrados en computadoras personales; estos equipos permiten al usuario tener un abanico de propósitos muy general y nada específico, al contrario que cualquier MCU presente en un sistema embebido o empotrado que, por definición, se construyen deliberadamente para servir a una aplicación específica.

Como los microprocesadores presentan, generalmente, un mayor número de pines que los otros (porque requieren de más conexiones con memorias, buses, periféricos...) y requerimientos más restrictivos en cuanto a velocidad, precisión, rendimiento... (que se traduce en una electrónica interna mucho más compleja [densidad de transistores]), los primeros suelen ser más caros, al menos si se busca integrarlos en un sistema completo (videoconsolas, controladores industriales o sistemas informáticos complejos, computadoras personales, de propósito general...).

Como consecuencia de sus aplicaciones y los sistemas hacia los que están enfocados, puede aventurarse que un microcontrolador debería de tener un ciclo de vida más prolongado que un microprocesador. Además, la complejidad de la aplicación puede significar tanto más instrucciones, como más incertidumbre al diseñar el sistema o sus procesos y modelar las tareas del μP . Pero generalmente, los microcontroladores disponen de librerías y compiladores específicos, así como los otros disponen de herramientas software concretas, que facilitan en gran medida la labor de su programación.

Echando un vistazo un poco más profundo, puede verse que los microcontroladores *modernos* suelen ser de 8, 16 ó 32 bits (máxima cantidad de información que pueden manejar [o pueden manejar los buses de datos y direcciones] al mismo tiempo), mientras que los microprocesadores acostumbran a fabricarse de 32 ó 64 bits.

B. Estructura

Atendiendo a las figuras Fig. 2. y Fig. 3. se puede deducir la diferencia más llamativa entre estos dos dispositivos; el MCU contiene no solo la CPU (al menos una), sino también otros componentes como memoria flash (RAM), ROM, periféricos y controladores de entrada y salida u oscilador interno, los cuales le otorgan funcionalidades adicionales respecto al microprocesador; básicamente, la sencilla y directa capacidad de reprogramarse y su adaptabilidad física (su estructura no hace que el sistema requiera de la integración de tantos componentes externos [aumentan el volumen], al contrario del μP). La arquitectura de los μC , en general, está basada en el modelo Harvard (el cual aprovechan para aumentar la velocidad de procesamiento de instrucciones y de acceso a los datos). Estas apreciaciones hacen del microcontrolador un

dispositivo más compacto y pequeño que el otro (bajo las mismas condiciones de operación).

C. Funcionalidad

En cierto modo, los microcontroladores son más lentos que los microprocesadores, en el sentido de que realizan, generalmente, menos instrucciones por segundo. Esto se debe a que trabajan con frecuencias de reloj mucho más pequeñas (típicamente, máximas de 300 MHz frente a 5 GHz...).

La diferencia de velocidades plantea una disparidad en la capacidad de memoria (de todo tipo): los μP necesitan mucho más espacio en memoria para almacenar y procesar todos los datos que se involucran en sus procesos a altas velocidades, por eso cuentan con mayores dimensiones (512 MB, 8 ó 32 GB de RAM, 128 a 2 TB de ROM...) respecto a los μC (512 bytes, 2kB, 256kB de RAM, 32kB a 2 MB de ROM...). Un efecto similar tiene esto en el consumo de potencia, el cual es mayor en el microprocesador, debido, entre otras razones, a la necesidad de conexión externa de la mayoría de los periféricos.

IV. CONCLUSIÓN

En este breve documento, se han logrado resaltar la mayoría de los aspectos importantes, características y generalidades típicas de los microprocesadores y microcontroladores, así como sus diferencias más notables. Sorprendentemente, más confusas pueden ser las diferencias de los tipos de dispositivos presentes dentro de estas dos clases, dado que la gran variedad comercial actual de ambos ocasiona un maremágnum de elecciones distintas para una sola aplicación en particular. Es interesante invertir algo de tiempo en reflexionar sobre qué microcontrolador quiere y debe usarse; en función de si la aplicación es de bajo costo o consumo (MSP430 de TI, STM32L0 de STMicroelectronics, LPC1768 de NXP Semiconductors, AVR de Atmel [Microchip]...), de si se requiere gran conectividad o de sistemas embebidos más complejos (PIC32 de Microchip o μC con núcleos ARM Cortex-M), de si interviene el procesamiento o control de señales de alta frecuencia (dsPIC) o de si se prefieren usar placas de desarrollo (STM32 Discovery, Xplained, PSoc, series Tiva de TI...), etcétera.

V. REFERENCIAS Y LECTURAS RECOMENDADAS

- David A. Patterson, John L. Hennessy, *Computer Organization and Design «The Hardware / Software Interface»*, 4a ed., 2012, Elsevier, ISBN 978-0-12-374750-1.
- Rajinder S. Kaler, *Microprocessors and Microcontrollers «Includes: Microprocessors-8085, 8086, & 80XXX Series, Microcontrollers-8051, ARM, AVR & PIC Series, Embedded Systems»*, 2a ed., 2014, IK International Publishing House, ISBN 978-93-82332-68-8.
- Krishna Kant, *Microprocessors and Microcontrollers «Architecture, Programming and System Design 8085, 8086, 8051, 809»*, 2a ed., 2012, ISBN 978-81-203-3191-4.
- Jonathan W. Valvano, *Embedded Systems «Real-Time Operating Systems for ARM Cortex-M Microcontrollers»* Vol.3, 4a ed., 2017, <http://users.ece.utexas.edu/~valvano/>, ISBN 978-1466468863.
- Bert van Dam, *PIC Microcontrollers «50 Projects for Beginners and Experts»*, 2a ed., 2008, Elektor International Media, ISBN 978-0-905705-70-5.
- Joseph Yiu, *The Definitive Guide to ARM Cortex-M3 and ARM Cortex-M4 Processors*, 3a ed., 2014, Elsevier, ISBN, 978-0-12-408082-9. Dave House, F. Faggin, H. Feeney, T. Hoff, S. Mazor, M. Shima, *Oral History Panel on the Development and Promotion of the Intel 4004 Microprocessor*, 2007, Computer History Museum, ref.: X4024.2007.