

ARM como Pivote del IoT

Por Gunnar Wolf



Gunnar Wolf es administrador de sistemas para el Instituto de Investigaciones Económicas de la UNAM y desarrollador del proyecto Debian GNU/Linux.

<http://gwolf.org>

● A lo largo de los años, fuimos testigos de cómo las arquitecturas rivales de cómputo pasaron de una gran gama de implementaciones luchando por la supremacía durante los 80 y 90 a una aparente monocultura. La primera década del milenio vio sucumbir a poderosas y prometedoras plataformas como Alpha, m680x0 y HPPA-RISC. Sparc y PowerPC redujeron su participación en el mercado a únicamente casos muy especializados. MIPS pasó de ser el último grito en lo relativo a hardware de alto rendimiento a ser una arquitectura para dispositivos de bajo consumo y bajo precio, como la mayor parte de los switches y puntos de acceso inalámbrico disponibles en el mercado.

CONSOLIDACIÓN DE UNA ARQUITECTURA REINANTE

Por varios años, parecía que la plataforma x86, definida por Intel como una serie de “parches” sucesivos al procesador 8086 de 1978, reinaría sin rivalidad en prácticamente todos los mercados. Computadoras de escritorio, portátiles, servidores. Hacia 2005, la plataforma x86 se erigió como clarísimo dominante. Y esto no por ser más potente o eficiente que su competencia, sino por tener una muchísima mayor base de usuarios ya instalada, por mantener una gran compatibilidad hacia atrás, y porque prácticamente todo el software de consumo masivo aparecía únicamente para éste.

Es muy sabido que esta compatibilidad hacia atrás no siempre es lo más conveniente técnicamente. En la década de 1980 nació RISC (Reduced Instruction Set Computing, cómputo con un conjunto reducido de instrucciones), primero como concepto teórico y poco más tarde implementado en varias arquitecturas de hardware (algunas de ellas mencionadas en el primer párrafo). Ya en esos años se veía que las principales arquitecturas disponibles tenían conjuntos de instrucciones demasiado complejos (recibiendo la denominación CISC, Complex Instruction Set Computing, cómputo con un conjunto de instrucciones complejo, en contraposición con la nueva tendencia).

Intel mismo intentó en dos ocasiones diseñar una nueva familia de procesadores que reemplazara a x86. La arquitectura RISC i860, lanzada en 1989 y abandonada unos cinco años más tarde, y la Itanium o IA-64, que se ha mantenido viva de manera formal

desde su debut en 2001, pero en la práctica y ante el abandono de prácticamente todos los fabricantes y proveedores de software, es una arquitectura cada vez más anecdótica.

ARM: JAQUE AL REY

Parecería que en este recuento de la gradual homogeneización de las plataformas de cómputo no hay mucho más que decir. Pero en los últimos años, se ha dejado sentir fuertemente un cambio drástico, y desde el segmento bajo del mercado.

La plataforma de hardware ARM no es bajo ningún concepto un nuevo jugador en el mundo del cómputo: su diseño original se produjo durante los años 1983 a 1985, al interior de la empresa Acorn, que buscaba desarrollar un sucesor para el popular CPU de 8 bits 6502 empleado en muchas computadoras de la época y, en particular, en la BBC Micro, plataforma de gran éxito en el Reino Unido. De hecho, ARM se manejó originalmente como acrónimo de “Acorn RISC Machine”, y una vez que la Acorn dejó de ser un hito reconocible en la mente de los usuarios británicos, para 1990 se independizó de la marca padre para denominarse “Advanced RISC Machines”.

Los procesadores ARM son un ejemplo para libro de texto de las diferencias y ventajas de RISC: si bien su conjunto reducido de instrucciones significa que, a una misma velocidad, pueden realizar mucho menos trabajo real que su competencia CISC, el número de transistores requerido para implementarlo resulta decenas de veces menor que el equivalente CISC de su época. Por ejemplo, cuando en 2007 se introdujo el ARM Cortex-A9, de 26 millones de transistores, los CPUs i7 de Intel excedían los 730 millones.

Menos transistores se traduce en muy distintas ventajas, todas ellas importantes para el cambio que hoy estamos observando:

- Fabricar CPUs ARM resulta más barato, dado que pueden usar procesos que requieren menor miniaturización y hay más fabricantes en condiciones de hacerlo (ARM no tiene ninguna fábrica, es dueño únicamente del diseño de los chips).
- La disipación de calor es menor. Un procesador

CISC actual requiere de enfriamiento activo, empleando masivos disipadores y ventiladores. La operación normal de un procesador ARM resulta mucho más aceptable para un equipo de perfil delgado, y más aún cuando está diseñado para llevarse en contacto con el cuerpo de su usuario.

- El consumo eléctrico es menor, permitiendo mucho mayor tiempo de operación a baterías. Naturalmente, el tamaño de los chips resultantes es menor. Mientras que un CPU Intel para equipos de escritorio mide unos 4cm. por lado, los procesadores ARM apenas y sobrepasan un centímetro cuadrado.

Los procesadores ARM siempre estuvieron presentes en nuestros dispositivos “menores”, pero dado que nuestra relación con estos nunca fue tan intensa y personal como en los últimos años, rara vez nos resultó claro. Al haber dado el gran salto cualitativo el mercado de los dispositivos móviles, la plataforma ha dado un salto tal que se volvió la primera respuesta al considerar cada vez más casos de uso, incluso ya en algunos casos entrando al segmento tradicionalmente fuerte de x86.

ARM es una arquitectura muy versátil; tienen tres perfiles de CPUs definidos. De más sencillo a más poderoso, “M” (para microcontroladores), “R” (para aplicaciones de tiempo real) y “A” (para aplicación).

COMPLICACIONES DE LA PLATAFORMA

La plataforma x86 tiene, sin embargo, una gran ventaja: la apertura y estandarización de las principales interfaces entre ésta y su sistema operativo. Para bien o para mal, desde la aparición de la IBM PC en 1981, todos los sistemas compatibles cuentan con un BIOS que se adhiere a una interfaz clara y pública; si bien esta ha sido extendida, mejorada y finalmente reemplazada — por UEFI, que a pesar de los puntos negativos que tenga, era una actualización largamente esperada — a lo largo de sus 35 años.

El BIOS funge de intermediario entre el hardware y el sistema operativo en sus primeras etapas (sobre todo en los primeros años de la PC; hoy en día la funcionalidad se limita casi en exclusiva a la inicialización del sistema). Al arrancar la computadora, se encarga de enumerar y realizar pruebas básicas de funcionalidad al hardware del sistema (el POST). Tras este paso, el sistema operativo puede lanzarse con conocimiento básico del hardware con el que operará.

En la arquitectura ARM, el BIOS sencillamente no existe. La razón para esto es que la compañía ARM funciona de forma distinta de Intel. La compañía no vende chips, no posee fábricas, sino que

licencia a terceros sus diseños como propiedad intelectual; los chip ARM son en realidad fabricados, entre muchas otras empresas, por AMD, Broadcom, Freescale, Texas Instruments, e incluso el mismo Intel.

En el mundo ARM, es muy común que los chips encapsulen no únicamente al CPU, controlador de memoria, controlador gráfico, y otros componentes esenciales, sino que prácticamente a todo el sistema de cómputo, inclusive la RAM y los controladores de dispositivos. A esta práctica se le conoce como “SoC” (System on a Chip); de hecho, si ven cualquiera de las computadoras basadas en ARM que han surgido en años recientes, uno de los primeros puntos que llama la atención es su relativa simplicidad. Y si acostumbran abrir, en claro ejercicio de un espíritu científico, sus teléfonos o tabletas, notarán precisamente lo mismo.

Las computadoras ARM, pues, no están normalmente concebidas para su expansión. Por sus características físicas, y por su bajo costo, es más bien un enfoque de un sistema para cada tipo de necesidad; desde el punto de vista en pro de la ecología suena nefasto, pero si requieren mayor poder de cómputo, la respuesta es cambiar de computadora. Además, la mayor parte de los fabricantes buscan ubicar sus equipos como “cajas cerradas”. ¡No es de su interés que yo pueda ejecutar mi software favorito en sus teléfonos en vez del “oficial”!

Ahora, dado que la computadora no requiere de la enumeración y prueba que realiza el BIOS, se ha optado por prescindir de este elemento. Es, pues, responsabilidad del sistema operativo saber qué dispositivos tiene y dónde se ubican. En el caso de Linux, el sistema operativo más común para entornos ARM, significa que debe construirse un núcleo ligeramente distinto para cada modelo de computadora. Este conocimiento del entorno se plasma en una estructura conocida como Device Tree (árbol de dispositivos).

Las plataformas Arduino y Raspberry Pi han resultado fundamentales, no por ser únicas o por ser las más económicas, sino por masificar el acceso a esta tecnología llevándolo al mercado educativo, y servir así como base para cientos de productos tangibles.

Adecuar Linux para su funcionamiento en una nueva sub-arquitectura no es una tarea trivial. Hay un fuerte empuje hoy en día para que los desarrolladores de hardware realicen este trabajo de adecuación con mayor colaboración y en consonancia con la comunidad desarrolladora primaria. Tengo plena esperanza en que, conforme se construye el ambiente de desarrollo para el afromado IoT, irá resultando claro a los fabricantes el valor de mantener la apertura ante los usuarios entusiastas. ☺