

Historias de OS

Para sistemas integrados en un μC



Por Stéphane Cnockaert

(Fuente : Airbus Industrie Toulouse)

No se trata ni de cocina ni de cine, sino de sistemas de explotación que no son privilegio exclusivo de los PC. Delante de la complejidad creciente de sistemas embebidos (integrados) y, teniendo en cuenta las misiones críticas de las que se encargan, la industria se apoya en sistemas operacionales fiables.

Hoy día podemos encontrar los microcontroladores en una gran cantidad de aplicaciones complejas, basadas en general sobre un único microcontrolador, que alcanzan todos los campos a la vez. A título de ejemplo, basta considerar un sistema de navegación de abordaje que asegura, sin fallar, el tratamiento digital de la señal (lectura de ficheros MP3, síntesis vocal), el control y la adquisición de datos provenientes de periféricos (subconjuntos GPS e información de tráfico vía radio), la gestión de una interfaz sin hilos (módulo Bluetooth), el almacenamiento de datos (tarjeta SD), y algo bastante evidente, el interfaz de usuario (pantalla gráfica táctil).

Fiabilidad demostrable

En los campos más especializados, tales como la aeronáutica, y más particularmente, los sistemas de control electrónico de vuelo, los microcontroladores han alcanzado un nivel

tal de miniatura que, a partir de ahora, parece razonable confiarles misiones críticas, tan críticas que el más mínimo fallo podría producir desastres que se cuantificarían en cientos de vidas humanas perdidas. Todo esto comenzó en ciertos aviones militares en los que se les instalaron para incrementar su maniobrabilidad y que serían difícilmente maniobrables en manual. Más tarde, se extendió a los aviones comerciales (Airbus A320 en 1987 – ¡ya!, Boeing 777 en 1994 – ¡solamente!) para limitar las tensiones sobre la estructura en las fases transitorias y, en menor medida, para reducir la resistencia aerodinámica (economizando combustible). Todas las aplicaciones informáticas para la aeronáutica deben estar certificadas por los estándares DO-178B (1992) y DO-248B (2001). Así pues, no es una casualidad que LynxOS-178 exista desde 1993, ya que se trata de una variante certificada de LynxOS, dotada de un módulo de programa que ayuda a efectuar pruebas

de certificación. *Nucleus*, en su variante DO-178B, es otro sistema operativo, dispuesto igualmente para la ayuda de la certificación FAA. ¡Fiabilidad demostrada!

El automóvil también

Otro lugar muy conocido donde se puede afirmar que la electrónica ha conseguido rellenar el hueco de las misiones críticas, es el sector del automóvil.

En la segunda mitad de los años 80 apareció la gestión denominada "programada", en la inyección de los motores de combustión (Bosch L-Jetronic, Honda PGM-FI, Marelli IAW), que sustituía la versión analógica aparecida a finales de los años 60 (Bosch D-Jetronic). Posteriormente, este movimiento se extendió a los sistemas de frenado ABS y, más recientemente, a la apertura optimizada de los airbags, el control de la estabilidad en la trayectoria, etc. Todas estas misiones son críticas: No tenemos más que imaginar las consecuencias fatales de la activación imprevista de uno de estos sistemas, cuando el vehículo circula a 120 km/h en una autovía repleta de tráfico.

Esta toma de conciencia no es de ayer, sino que, desde 1993, ya fue puesta en marcha la norma OSEK (que viene de *Offene Systeme und deren Schnittstellen für die Elektronik in Kraftfahrzeugen* en alemán, *Open Systems and their Interfaces for the Electronics in Motor Vehicles* en inglés o *Sistemas Abiertos in sus interfaces para la Electrónica en Vehículos a Motor* en español), un consorcio formado por todos los fabricantes de automóviles y de equipamientos alemanes, asociados a un departamento de la Universidad de Karlsruhe. En 1994 se produjo la unión con los trabajos similares de los fabricantes franceses (Renault y PSA), lo que ha hecho que la norma se denomine hoy día OSEK/VDX (para, *Vehículo Distribuido e Ejecutivo*), una norma que ha introducido el lenguaje OIL para el establecimiento y la descripción de los servicios (OSEK *Implementación Lenguaje*). Así pues, diferentes OS (*Operating System*, es decir, Sistemas Operativos), tales como PICOS18 y *Nucleus*, pueden ser revestidos de una capa que responde a la norma OSEK/VDX. Debemos señalar que este estándar va un poco más lejos que la fiabilidad de una aplicación sobre un μC aislado en la medida, donde también están involucradas las comunicaciones entre los procesadores, las redes y el tratamiento de datos distribuido.

¿En casa de Elektor?

A menudo es el enemigo quien nos hace mejores. Habiendo realizado innumerables proyectos de gestión, basados en microcontroladores potentes como los de la serie ARM7, ¿cómo resistirse a las ganas de realizar montajes, sin "partir de cero", sino a partir de un sistema operativo (SO) fiable y documentado?

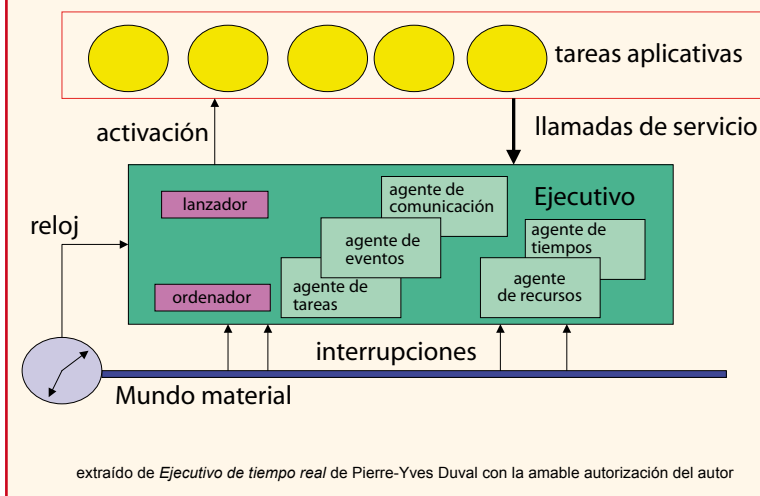
Sobre todo cuando ciertos SO proporcionan, no solamente el control sobre ciertos periféricos tales como USB, tarjetas de memoria (SD y otras), sino también el protocolo TCP/IP, e incluso el propio interfaz gráfico de usuario.

Otras nociones especializadas, tales como núcleo, ejecutable, programador de tiempos, sistema propietario, tiempos compartidos, la atomización, las prioridades, la reentrada, las pilas, la cola de espera,... nociones que condicionan las aptitudes en tiempo real (y sobre las cuales será necesario que Elektor vuelva, si sus lectores lo demandan, así que esperamos vuestros mensajes sobre este tema, dirigiéndoos a stephanecknockaert@hotmail.com), son en realidad ventanas abiertas a una rica conectividad que nos motiva en nuestra marcha. Dejemos de reinventar la rueda.

¿Nos seguirán nuestro motivado público, profesional o aficionado apasionado?

Programar el lenguaje C en lugar de ensamblador es

Estructura de un ejecutivo



ya una evolución. Pero abandonar lo "hecho a mano" para pasar a trabajar sobre la concepción de un conjunto impuesto, ¿es una evolución o una regresión? ¿Todo esto para recuperar librerías y controladores? ¿Vale la pena el reto?

En un futuro muy próximo

Para abrir boca, nos ha apetecido el compartir con nuestros lectores la experiencia de Dirk Krämer, que nos ha descrito la situación actual de LAOS, su pequeño sistema operativo (no propietario) para los microcontroladores ARM7. Entremos pues en su "cocina de hechicero" e intentemos salir llenos de nuevas ideas y sanos de espíritu!

(071170-I)

Enlaces en Internet

Proponemos a nuestros lectores, en complemento de este artículo, una recopilación abundante de enlaces que tratan sobre este tema. Merece la pena su visita!

Sincronización por eventos sincrónicos

Fases temporales del caso de un evento

