

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO



FACULTAD DE INGENIERÍA

SISTEMAS OPERATIVOS

(1554)

Primer Proyecto:

Revisión de MiComputer

FECHA DE ENTREGA: 30 DE AGOSTO DEL 2018

SEMESTRE: 2019-1

ALUMNO: CRISTIAN VALERIANO BARRIOS

PROFESOR: WOLF ISZAEVICH GUNNAR EYAL

Jupiter Ace

La Jupiter Ace fue una computadora personal lanzada al mercado en el año 1982, por la extinta empresa *Jupiter Cantab*.

Su popularidad se debió a precio económico, con lo que estaba al alcance de cualquier entusiasta de la computación, pero además destacó por implementar el lenguaje de programación *FORTH*, en lugar de *Basic*, el cual era el lenguaje predominante en el mercado.

Lo interesante de lo anterior mencionado es que el manual de uso del equipo, introducía a los usuarios al aprendizaje del lenguaje *FORTH*, y con ello a un paradigma diferente: mientras un programa en *BASIC* realizaba operaciones algebraicas y lógicas a través de expresiones, *FORTH* lo hacía manipulando la memoria *stack*.

Otra característica importante de dicho lenguaje era la capacidad del usuario de definir nuevas órdenes para utilizarlas posteriormente.

Y por si lo anterior no hubiese sido suficiente, la Jupiter Ace fue diseñada por parte del equipo que trabajó en las populares *ZX Spectrum* y *ZX81*, ambas de la empresa *Sinclair*, por lo que su apariencia era bastante similar, e inclusive era compatible con los paquetes de ampliación de memoria existentes para la *ZX81*. Muchos beneficios a un precio accesible.

Lo que más me llama de atención de este artículo es que, entre sus diversas especificaciones técnicas, resalta la característica de poder expandir su memoria de 3KB hasta 51 KB. Este tipo de expansiones siguen siendo una tendencia en nuestros días principalmente en dispositivos móviles, pero la diferencia en escala es abismal, sin que ello demerite el esfuerzo que los diseñadores hicieron con tales limitaciones tecnológicas de la época.

Gráficos con vida

En los años 80 existían principalmente dos métodos para generar gráficos animados por computadora. El primero consistía en desplegar imágenes de forma secuencial a una velocidad razonable que generara la ilusión del movimiento. El segundo método se trataba de dibujar un caracter en pantalla, borrarlo y dibujar el siguiente. Implementar lo anterior en lenguaje Basic resultaba en un programa poco eficaz a la hora de realizar el procedimiento a la velocidad suficiente para que la animación fuera fluída.

La resolución de ese problema se encontró en escribir dichos programas a bajo nivel, en ensamblador, para así manejar los datos directamente en memoria y mapearlos a gran velocidad en la pantalla.

Para ello se idearon objetos denominados *sprites*, los cuales también tenían la capacidad de crear la ilusión de transitar en diferentes planos independientes, y con ello se emulaban gráficos complejos en tres dimensiones. Cada sprite se compone de matrices de píxeles, y la memoria almacenaba la información del comportamiento de cada uno de ellos.

De esta forma, una computadora simple podía realizar los cálculos necesarios para manipular la velocidad, color, tamaño y desplazamiento de cada sprite, a través de cada píxel.

Es increíble como esta implementación representó una revolución y un importante precedente para la generación de gráficos de manera eficaz, exprimiendo recursos y capacidad de cómputo que en nuestros días parecerían insignificantes.

Artículos consultados en la revista MiComputer, fascículo número 8, volumen I, publicado en 1984 por Editorial Delta S.A., Barcelona.