

Motores gráficos

Ensamblador ha demostrado ser una herramienta capaz de realizar cualquier tipo de tarea haciendo uso de uno de los conceptos más importantes de la programación: solucionar un problema dividiendo en subproblemas. Solo con la existencia de herramientas como los saltos de memoria ya es posible realizar una infinidad de tareas (condicionales, bucles, funciones, etc.), y con la combinación de diferentes tipos de saltos es posible solucionar casi cualquier problema.

Además, una implementación aún más flexible son las interrupciones, con las cuales es posible guardar el estado actual del sistema para realizar otras tareas y luego continuar la ejecución principal. Esto permite desarrollar sistemas mucho más complejos y también permite guardar rutinas de uso común para facilitar su implementación.

Una de las interrupciones que permitió una mayor accesibilidad para los usuarios es el modo de video (int 10h), con el cual es posible mostrar texto en pantalla e incluso dibujar figuras con colores variados. El funcionamiento de este modo es simple: se indica una posición de la pantalla y se muestra un pixel coloreado en esa parte.

Al hacer uso de las diversas interrupciones de este modo es posible crear figuras bastante complejas a partir de otras más simples. Por ejemplo, es posible crear líneas con muchos puntos seguidos, o cuadrados haciendo varias líneas juntas, etc.

Todo lo que es visible desde un monitor ha sido creado utilizando este modo del procesador en conjunto con otras herramientas que proveen los diversos tipos de monitor que existen. Teniendo en cuenta esto, todas las ventanas, íconos, punteros, etc. son simplemente llamadas a subrutinas de dibujado.

Muchos lenguajes de alto nivel en la actualidad cuentan con soporte para herramientas gráficas (JavaFX, TKinter, Swing, etc.), pero el fundamento detrás del manejo de ventanas reside principalmente en el establecimiento de límites abstractos (por ejemplo $-5 < x < 5$ y $-3 < y < 3$ son límites de un rectángulo) y mostrando esos límites visualmente por medio de líneas.

Ahora bien, dentro de las ventanas pueden existir más elementos visuales para generar algún tipo de interacción con el usuario. A medida la tecnología fue creciendo y permitiendo operaciones más complejas se inició el desarrollo de aplicaciones de ocio (como los videojuegos).

Los videojuegos inicialmente eran muy simples, mostrando únicamente imágenes poco detalladas; sin embargo, poco a poco se introdujeron pantallas con mayor resolución que permitieron mucho mayor detalle para los contenidos de los videojuegos. Esta mayor capacidad de detalle permitió el salto de 2D a 3D.

Los videojuegos en 3D realmente carecen de una característica fundamental del “espacio”: **la profundidad**, por lo que para emular la profundidad se hace uso de la perspectiva. Para el procesador no existe diferencia entre 2D y 3D visualmente hablando (ya que la cantidad de operaciones a realizar sí es mayor), simplemente sigue siendo un conjunto de puntos que representan una imagen.

Realizar manualmente la asignación de todos esos puntos resultaría en una pérdida de tiempo considerable, por lo que se han desarrollado programas que ejecutan todas las interrupciones necesarias para el renderizado de las imágenes: los motores gráficos.

En esencia estos motores siguen realizando las operaciones básicas originales, pero cuentan con herramientas que facilitan el diseño y visualización de lo que quiere plasmarse en pantalla, además de ocuparse de los cálculos complejos para la interacción con el espacio.

Resulta bastante impresionante el hecho que sea posible emular tres dimensiones a partir de simples líneas, pero esto tomó muchos años en poder desarrollarse de manera aceptable (considerando que aún en la actualidad hay mucho espacio para mejoras).

Hay que considerar que gran parte del trabajo de codificación se ha simplificado con los lenguajes de alto nivel, sin embargo, en el fondo continúa siendo código de ensamblador. Ensamblador no ha desaparecido, pero se han desarrollado herramientas para facilitar su manipulación y así permitir desarrollar operaciones más complejas con facilidad.