



Universidad Nacional Autónoma de México

Proyecto 01 - Reseña MiComputer

Curso: Sistemas Operativos

Profesor: Dr. Gunnar Eyal Wolf Iszaveich

Autor: Chávez López B. Alejandro

Cuenta: 421112601

Grupo: 08

September 19, 2025

En este ensayo se abordan los temas presentados en la revista **MiComputer**, publicación que acercó el mundo de la computación a un público general en la década de 1980. En mi caso, elegí el fascículo número 21, el cual incluye como artículo central la reseña de la computadora **Osborne-1**, además de apartados dedicados a un dispositivo portátil muy interesante **Microwriter** y a programas de corrección de textos **Magic Spell**.

1 Microwriter

En la sección de *Insights* se habla acerca de un procesador de texto portátil llamado **Microwriter**, el cual era un dispositivo portátil que tenía como particularidad un dispositivo de entrada tipo mnemónico. Este consistía en seis teclas, las cuales formaban letras al ser presionadas en un orden en particular.

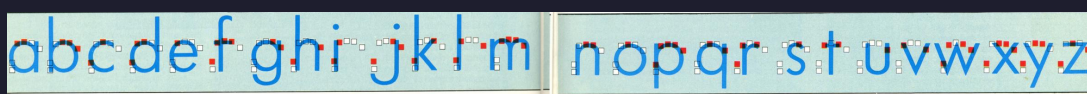


Figure 1: Ilustración de las entradas posibles usando las 6 teclas.

Esto me recordó a las máquinas de estenotipia (o estenotipos), que son teclados que, de igual manera, tienen menos teclas que un teclado convencional, pero que al presionar varias al mismo tiempo forman sílabas o palabras. Son muy utilizados por reporteros de juicios o subtituladores en vivo por ser mucho más rápidos, aunque también mucho más difíciles de usar que un teclado normal.



Figure 2: Máquina de estenotipia.

En cualquier caso, este aparato me resulta especialmente interesante por varias cuestiones.

Su procesador, el **CDP1802**, fue (según lo investigado) el primer procesador CMOS, y al leer más acerca de las ventajas de los procesadores CMOS me sorprendió saber que este proceso de fabricación permitió que no hubiera una frecuencia mínima obligatoria. Se podía bajar mucho la velocidad de reloj (o incluso detenerlo), tanto para ahorrar energía —lo cual le daba una ventaja enorme frente a otros procesadores portátiles en cuanto a

batería— como para poder reanudar el estado de la computadora y seguir escribiendo. En general, el hecho de incluir un procesador de características similares a los de otras computadoras de su época refleja la relevancia de la llamada **Ley de Moore**, ya que tuvieron que existir avances muy significativos para crear un procesador tan compacto y eficiente para usarse en equipos portátiles.

También contaba con interfaces de E/S muy interesantes que le permitían ser utilizado como una terminal convencional en otras computadoras o procesadores de texto, así como comunicarse con impresoras. Esto me recordó al tema de procesos e hilos, ya que se comportaba como un proceso de entrada conectado a un sistema más complejo en el modo terminal.

De forma general, este dispositivo me pone en perspectiva cómo hemos avanzado en cuanto a dispositivos móviles y microprocesadores, ya que actualmente podemos encontrar computadoras completas en formatos incluso más pequeños, como la serie de **Raspberry Pi**, que utilizan muy poco espacio, son mucho más eficientes energéticamente y pueden correr sistemas operativos de escritorio completos.

2 Magic Spell

En la sección de *Software* se habla de programas de procesamiento de lenguaje natural, lo cual me pareció muy relevante en el contexto actual, ya que los programas mencionados y los actuales han cambiado de manera significativa.

Se menciona que, en el contexto del lenguaje natural, las computadoras, incluso en esa época, ya podían resolver casos donde la sintaxis de una oración no era adecuada. Se pone como ejemplo los errores de sintaxis durante la compilación de un programa. Esto me recordó a los ejercicios que hacíamos en la clase de **Lenguajes Formales y Autómatas**, donde, dadas ciertas reglas como un alfabeto, símbolos de inicio y final, etc., se podían generar palabras válidas para un lenguaje. Gracias a esto fue posible la creación de programas que revisaban documentos para encontrar errores de escritura u ortografía en lenguajes como el inglés, en este caso.

En la explicación del funcionamiento de estos programas me llamó la atención que se mencionan dos cosas que han cambiado mucho desde entonces. Para poder guardar las palabras, se usaban diccionarios almacenados en disco que podían llegar a pesar más de 150 kilobytes, lo cual era una gran limitante en esa época. Por ello se implementaron algoritmos y técnicas para reducir el tamaño de los diccionarios lo más posible. En la actualidad, eso es algo que fácilmente podemos almacenar sin problemas. También se decía que, de manera ideal, el programa debería revisar cada palabra tan pronto se escribiera en el documento (comparándola con el diccionario). Pero como era muy tardado y costoso computacionalmente, se optaba por guardar las palabras en un arreglo y, al final, compararlas con el diccionario. Esta técnica reducía los tiempos de ejecución; se ponía como ejemplo los 4 minutos en el caso de **VizaSpell**. Hoy en día, en cambio, programas como **MS Word** o **Google Docs** revisan cada palabra tan pronto como se escribe, detectando problemas de escritura, ortografía e incluso de gramática, como cuando una palabra no está bien conjugada o se mezclan plurales y singulares en una misma oración.

Esta sección termina con una idea cómica:

```
1 COMMAND > GENERATE ARTICLE. LENGTH 1200 WORDS, BEGIN
```

Algo que en esa época parecía imposible, pero que hoy es muy real con herramientas como **ChatGPT** o **Gemini**, que cumplen con todo lo que se describe: corrigen faltas ortográficas, errores de escritura, estilo y hasta generan (con muchas consideraciones de lo que consideramos "generar") texto.

3 Osborne-1

En la sección *Hardware Focus* de esta edición se habla de la computadora **Osborne-1**. Esta fue, según el artículo, la primera microcomputadora completamente contenida (lo que incluso se puede considerar como una especie de laptop). Aunque no era muy liviana, sí era portátil para los estándares de la época. Algo que me llamó mucho la atención es que su sistema operativo **CP/M** permitía hacer programas para cualquier computadora, sin importar la configuración de hardware, siempre y cuando usara el mismo sistema operativo. Esto me parece fundamental para la programación actual, ya que en la mayoría de los casos podemos realizar programas que, una vez compilados (o ejecutados por un intérprete), pueden usarse en cualquier máquina. En esos años, esto no era lo normal.

Durante alguna clase se habló de las llamadas al sistema, las cuales variaban según la arquitectura del procesador y el sistema operativo, pero que a nivel de programación no hacen diferencia, ya que los programas en espacio de usuario simplemente piden un servicio al kernel sin acceder directamente al hardware. Me interesó ver cómo este nivel de abstracción ya se implementaba desde esos años, y fue un gran atractivo para los programadores de esa generación.

Se menciona que, en el mercado estadounidense, se adoptó la **IBM Personal Computer**, que era una máquina de 16 bits basada en el procesador Intel 8088 y que corría el sistema **PC-DOS**. Como competencia, Digital Research lanzó dos versiones del sistema CP/M, entre las cuales destacó la versión **Concurrent**, que permitía **multiprogramación multiusuario**. Esto significaba que varios usuarios podían interactuar de manera simultánea con un mismo equipo y que múltiples programas podían permanecer activos en memoria al mismo tiempo. Esto implicaba un mejor aprovechamiento de la CPU, que ya no quedaba inactiva durante las operaciones de entrada/salida, y facilitaba el trabajo colaborativo, ya que varios usuarios podían ejecutar tareas diferentes sin necesidad de esperar turnos.

Si bien la empresa responsable de este modelo eventualmente quebró, me sorprenden las innovaciones que introdujeron, tanto con su hardware como con sus sistemas operativos, ya que fueron pioneros en implementar características que hoy damos por sentadas en los sistemas modernos.

Conclusión

En este fascículo de **MiComputer** pude ver cómo los avances en hardware y software de esa época marcaron la base de muchas ideas que hoy damos por sentadas. Al leer sobre la Osborne-1, el Microwriter y la sección Magic Spell, noté la conexión entre esas innovaciones y lo que hemos trabajado en clase: multiprogramación, abstracción del hardware, eficiencia en recursos e incluso (en cierta medida) al procesamiento de lenguaje natural. Lo que más me gustó de la revista fue esa mirada histórica que permite entender, desde la perspectiva de los usuarios y entusiastas de aquel tiempo, cómo fue evolucionando la computación.