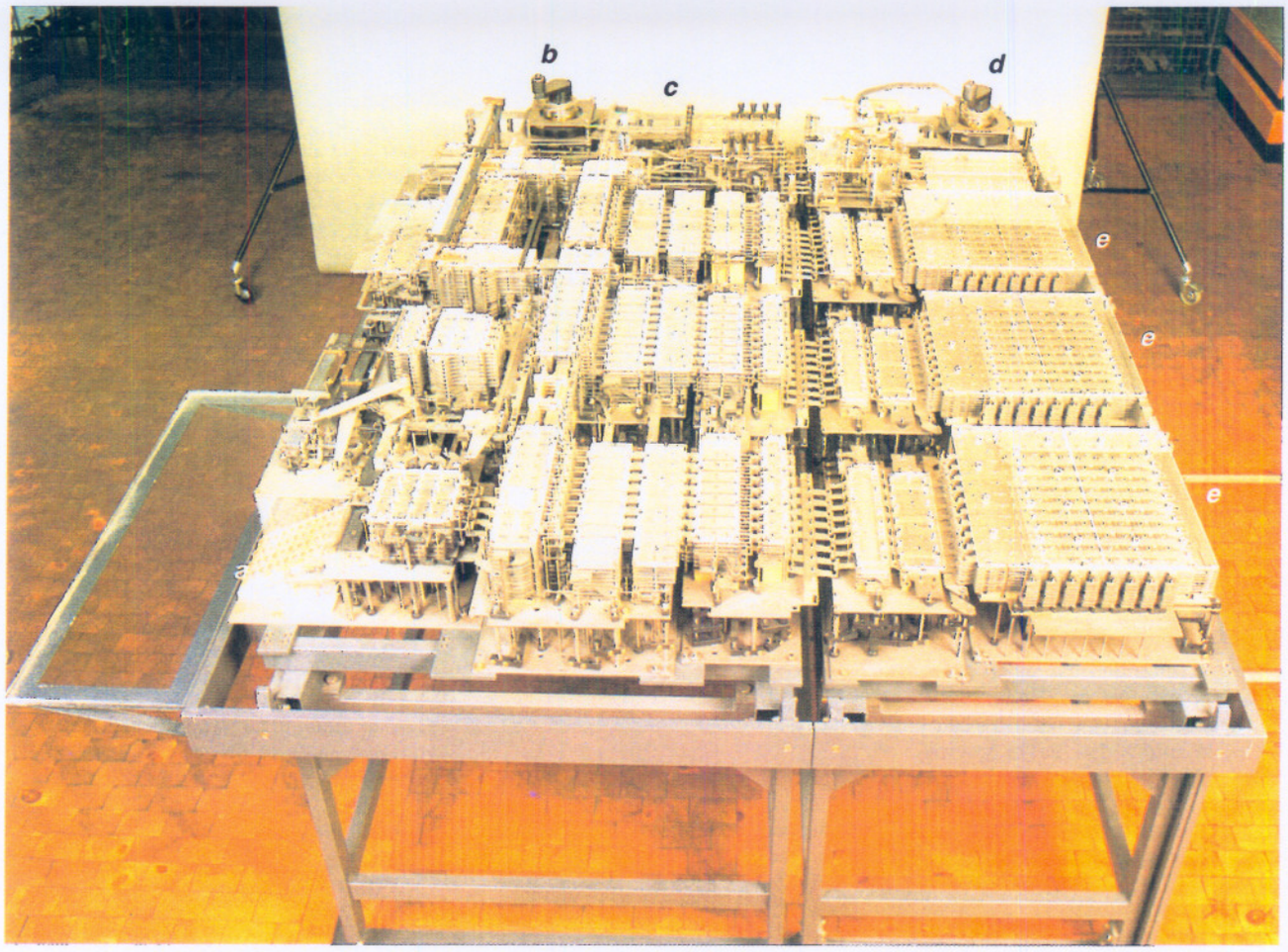


Los ordenadores de Konrad Zuse

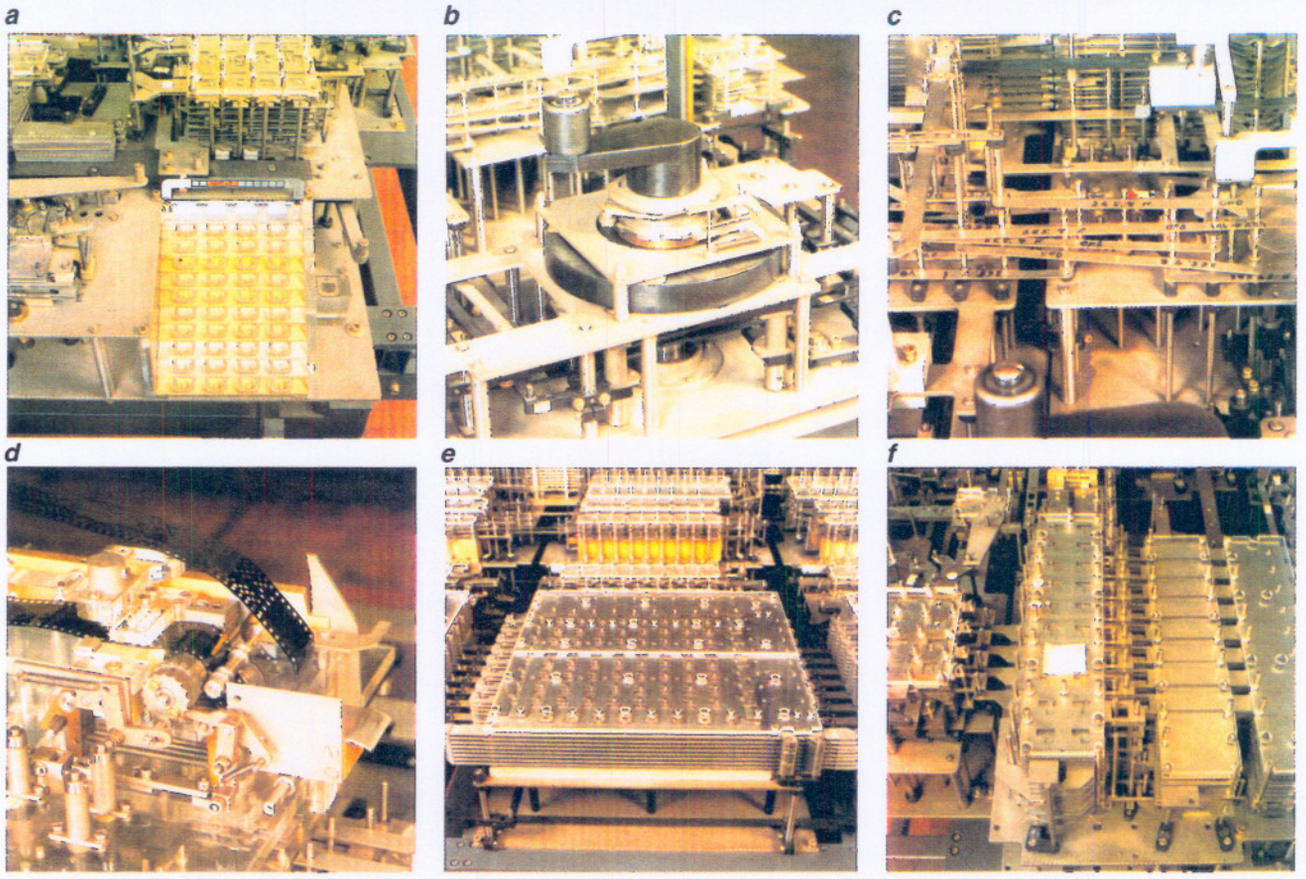
Entre 1936 y 1941 este famoso inventor alemán construyó en Berlín dos máquinas de cómputo a las que llamó Z1 y Z3. Aunque el diseño lógico de ambas era muy similar, creó la primera con componentes mecánicos y la Z3 con relés telefónicos

Raúl Rojas

Desde hace años se viene discutiendo sobre la fecha y el lugar en donde se inventó el ordenador universal. Ciertas instituciones de los Estados Unidos celebraron en 1996 las bodas de oro, en conmemoración de que se cumplían 50 años de la presentación pública del ENIAC (*Electronical Numerical Integrator and Computer*). Esta máquina fue en su época el ordenador más rápido. De ahí que la prensa la considerase durante muchos años el primer ordenador del mundo. La verdad es que los propios norteamericanos han ignorado el trabajo de otros compatriotas, así John Atanasoff (1903-1995), o no han otorgado la debida importancia a las invenciones de extranjeros, como ha sucedido con Konrad Zuse (1910-1995).

En este artículo me propongo analizar la estructura lógica de las máquinas de cómputo creadas por Zuse. Me ocuparé, sobre todo, de la estructura de dos de sus primeras creaciones, la Z1 y la Z3, que llama la atención por su sorprendente actualidad cuando se las compara con la estructura de procesadores construidos hasta hace pocos años. Para realizar este trabajo, conté con la ayuda directa de Zuse. En 1995 me entregó, fotocopiada, la documentación que había adjuntado a la solicitud de patentes en los años de la segunda guerra mundial.

Quienes estén familiarizados con la estructura de ordenadores reconocerán de inmediato en la arquitectura de las máquinas de Zuse conceptos que, por su importancia, se han convertido en poco menos que dogma. Me refiero, por ejemplo, a la separación del procesador de la memoria, la utilización del sistema binario, etcétera. Para ponderar la visión de su autor, téngase presente



1. EL ORDENADOR MECANICO Z1. Se reconstruyó bajo la dirección del propio Konrad Zuse, basándose en algunas notas y en la todavía portentosa memoria de su provecto inventor. La máquina original se destruyó en la guerra. De la talla de un piano de cola, Z1 consta de varias estructuras metálicas que pueden separarse para el transporte. Los movimientos de los componentes se transmiten de una estructura a otra a través de acoplamientos mecánicos. El operador puede alimentar un número decimal en el ordenador tirando de las piezas metálicas (*a*) que corresponden a las cifras decimales, el exponente y el signo del número. Para mostrar un resultado numérico se activan pequeñas láminas. A un lado de la unidad de entrada y salida se encuentra el mecanismo que transforma un número decimal en otro binario, y viceversa. Por medio de una manivela (*b*), que se puede operar manualmente o con un motor, se transmiten los movimientos necesarios a todas las partes mecánicas; cada giro de la manivela corresponde a un ciclo de máquina. Muchas barras (*c*) transmiten los movimientos de la manivela a otras partes de la máquina, a tenor de la instrucción que se esté ejecutando. El programa se lee de una cinta perforada de 35 mm, como las usadas en cinematografía. La unidad de lectura (*d*) hace avanzar la cinta perforada un puesto cada vez que se lee una instrucción, decodifica la instrucción e inicia las operaciones correspondientes. Los tres bloques idénticos (*e*) de la derecha constituyen la memoria, capaz de almacenar 64 números: un bloque es para el exponente y dos para la mantisa. Cada bloque contiene 7 bits. Para leer un número de una dirección de memoria se activan los elementos mecánicos correspondientes. Los datos fluyen hacia el procesador utilizando un componente (*f*) que Zuse consideraba similar a las vías de un ferrocarril. Pese a que en la reconstrucción se dispuso de instrumentos de precisión, la máquina se encallaba a veces. Zuse adivinaba en seguida dónde residía el fallo, dónde bastaba oprimir con el dedo para que la operación continuara. La réplica que mostramos, terminada en 1989, se encuentra en el Museo Alemán de la Técnica en Berlín.

que Zuse avanzó tales conceptos en 1936, el año en que tuvo listo el prototipo de la memoria de la Z1.

Por esa misma fecha, Alan M. Turing (1912-1954) preparaba el primer borrador de una artículo que revolucionaría la matemática; en él se formaliza el concepto intuititivo de "computabilidad". Turing definió para ello una máquina ideal: sólo los problemas numéricos resolubles con esta máquina pueden denominarse problemas computables. Una máquina de Turing consta de una banda infinita que almacena ceros y unos (una memoria) y un pequeño procesador capaz de leer y modificar una sola cifra binaria a la vez. Turing demostró que esa máquina tan primitiva podía operar como máquina universal, es decir, como un autómata capaz de simular cualquier otra máquina de Turing. Para ello se debe preparar la banda-memoria con dos tipos de contenido: los datos del problema a resolver y lo que hoy llamaríamos el programa para realizar el cálculo.

Por tanto, en 1936 no surgió el ordenador en el sentido material y moderno del término, pero sí quedó formulado claramente su concepto. En el mundo de la teoría se había inventado el ordenador universal.

Zuse no era matemático. Tampoco se ocupó nunca de definir la computabilidad de manera teórica. No conocería los escritos de Turing hasta muchos años después de haber realizado su propio trabajo. A él, que procedía del ámbito de la praxis, lo que le interesaba era facilitar el cálculo de largas cadenas de operaciones. Gracias a su formación en ingeniería y a su habilidad para estructurar sistemas complejos partiendo de com-

Notación en coma flotante

El número decimal 18,73 es una forma abreviada de expresar

$$1\times10^{1}+8\times10^{0}+7\times10^{-1}+3\times10^{-2}.$$

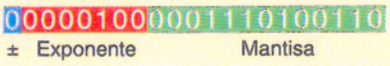
De manera similar, el número binario 10,11 se interpreta mediante la descomposición

$$1\times2^{1}+0\times2^{0}+1\times2^{-1}+1\times2^{-2}.$$

Cualquier número (excepto el cero) se puede expresar en la forma $\pm a\times2^{b}$. El exponente b es un número entero y la mantisa a un número entre 1 y 2. Un ejemplo podría ser $a=1{,}11$ y $b=2$. El número binario 1,11 expresado en forma decimal es 1,75, ya que $1\times2^{0}+1\times2^{-1}+1\times2^{-2}=1{,}75$. Esa es la razón por la que en este ejemplo $a\times2^{b}$ representa al número 7 que es igual a $1{,}75\times2^{2}$.

Zuse reservó siete bits para el exponente (*de color rojo en el diagrama*). Con siete bits pueden representarse los enteros de 0 a 127. Sin embargo, Zuse decidió utilizar el "complementario de dos", representación en la que el bit no tiene en la posición más alta un peso de $64=2^{6}$ (como sería el caso en una representación binaria normal) sino de –64. De ese modo se puede, pues, representar con siete bits todos los exponentes enteros desde –64 hasta –63.

Zuse utilizó para la mantisa 15 bits (*verde*) y un bit adicional para el signo (*azul*). Todas las cifras se normalizan antes de retenerse en la memoria; la única cifra binaria antes de la coma será 1. Este bit no necesita almacenarse en memoria (siempre es 1); por eso se requieren sólo 14 bits para guardar una mantisa en la memoria. Cuando el procesador lee la mantisa de la memoria, basta agregar la cifra 1 antes de la coma decimal.



La cifra mostrada en nuestro ejemplo es positiva (el bit de signo es 1), posee el exponente 100 binario (igual al 4 decimal) y la mantisa 1,000 111 010 001 10 (= 1,1136 en decimal, utilizando sólo cinco cifras). El número decimal representado por este exponente y mantisa es entonces $1{,}1136\times2^{4}=17{,}818$.

Hay un problema con este tipo de representación: el número cero no puede expresarse con la notación normalizada, ya que la mantisa nunca puede ser igual a cero. Por eso Zuse decidió adoptar una convención especial: cualquier número de la forma $a\times2^{-64}$ se interpreta como cero. En Z3 hay los circuitos necesarios para identificar este caso especial y acometer los cálculos correctos. Otro caso especial es el de cualquier número con exponente +63, que se interpreta como infinito o menos infinito, de acuerdo con el signo de la mantisa.

Para ponderar la visión anticipadora de Zuse, basta comparar sus convenciones numéricas con la norma para operaciones de coma flotante definida por el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE). La notación del IEEE es muy similar a la notación semilogarítmica utilizada por Zuse (con ciertas diferencias en la codificación de los exponentes y en los casos especiales considerados).

ponentes simplísimos reparó en la posibilidad de construir autómatas programables.

Su primer intento en esta dirección fue la Z1 (*véase la figura 1*), un aparato casi totalmente mecánico, capaz de ejecutar las cuatro operaciones aritméticas (suma, resta, multiplicación y división) en cualquier secuencia y con números almacenados en una memoria. Cuando terminó la Z1 en 1938, comprobó que los componentes mecánicos no eran del todo fiables y decidió adoptar otra técnica, la de relés electromagnéticos.

Tras algunos ensayos con una máquina híbrida, la Z2, Zuse construyó un nuevo prototipo, la Z3, que en 1941 ya operaba. Desde el punto de vista lógico remedaba la Z1. Los algoritmos numéricos y la codificación de los números eran los mismos. En cierta medida Z3 venía a ser una Z1 fiable. En sólo cinco años el inventor berlinés había hecho realidad su intuición visionaria de 1936. Considerado desde la perspectiva de los proyectos paralelos en Norteamérica, que contaron desde el principio con recursos ilimitados (piénsese en el ENIAC), lo logrado por Zuse resulta todavía más admirable.

Los prototipos originales de ambas máquinas se destruyeron durante la segunda guerra mundial. Zuse reconstruyó en 1966 la Z3 y, en el bienio de 1987-1989, la Z1. Ambas reconstrucciones, que ahora operan parcialmente, se exhiben en las exposiciones de los museos de ciencia de Berlín y Munich.

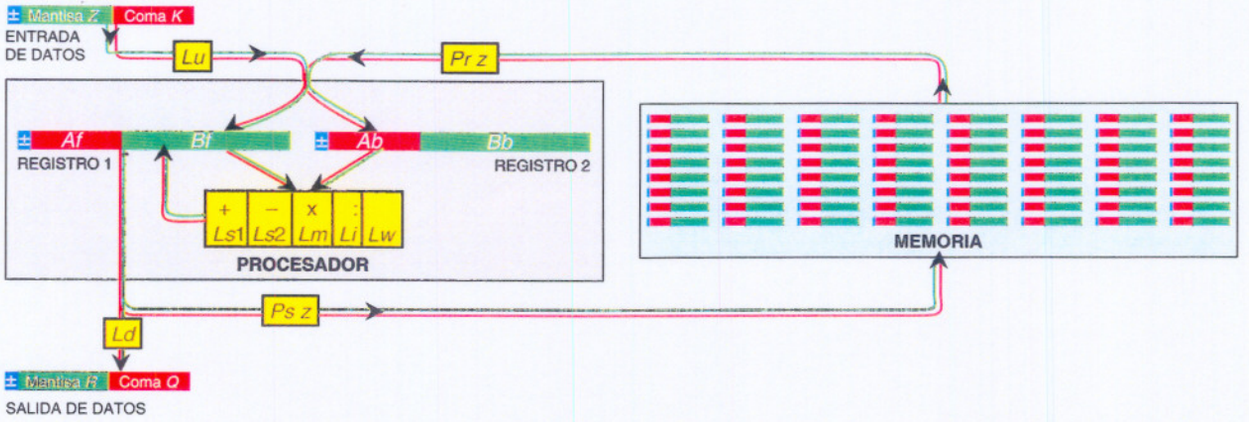
La primera decisión adoptada por Zuse apunta ya hacia la estructura de los ordenadores modernos: sus máquinas operan en sistema binario. En la representación binaria sólo existen las cifras 0 y 1. Cada componente mecánico o eléctrico capaz de adoptar dos estados distintos puede servir para representar una cifra binaria. En el caso de la Z1 se utilizan finas barras metálicas que proceden hacia adelante o hacia atrás (a lo largo de la barra). En el caso de la Z3 se utilizan relés electromagnéticos: 600 para el procesador y 1400 para la memoria (*véase la figura 2*).

La elección de la base 2, que hoy nos parece obvia y natural, no era la única opción discutida en aquella época. Los computadores Mark I y ENIAC, construidos pocos años después en los EE.UU., trabajaban con una representación decimal. El ENIAC, máquina electrónica, utilizaba una cadena de 10 tubos de vacío para codificar cada dígito. Sólo el tubo que correspondía a la cifra representada se apagaba; todos los demás permanecían encendidos.

La representación binaria, sin embargo, es mucho más fácil de manipular, lo mismo con componentes mecánicos que electrónicos. Además, argumentaba Zuse, en el ordenador los números están sólo en relación "consigo mismos" y el operador no necesita poder seguir cada paso de un cálculo. Basta con que el resultado final de la cadena de operaciones sea correcto y se pueda transformar al sistema decimal.

La memoria de la Z3 puede almacenar 64 números de 22 bits. Igual que las memorias modernas, posee una estructura repetitiva bastante sencilla de construir. Pero Zuse optimizó el procesador con muchas ideas novedosas y por eso es mucho más complejo. El procesador puede eje-

Programación de Z3



Desde el punto de vista del programador, el procesador de la máquina Z3 contiene dos registros de coma flotante (R1 y R2). El programador no necesita saber que Zuse los llamaba registros *f* y *b* y que además en el procesador existe un registro temporal adicional llamado *a* (*véase la figura 4*). La primera instrucción de un programa que lee de la memoria carga el número leído en el registro R1 (si es un número que viene del teclado decimal, lo carga después de convertirlo en número binario). Cualquier instrucción de lectura posterior se carga en el registro R2. Esto continúa mientras no se borre el registro R1.

Z3 se programa con las nueve instrucciones siguientes (en la primera columna tenemos el nombre que Zuse le dio a la instrucción, en la segunda su codificación binaria):

Entrada y salida:

Lu	01 110000	lee del teclado
Ld	01 111000	despliega el resultado

Memoria:

Pr z	11 $z_6z_5z_4z_3z_2z_1$	carga de la dirección *z*
Ps z	10 $z_6z_5z_4z_3z_2z_1$	almacena en dirección *z*

Aritmética:

Ls_1	01 100000	adición
Ls_2	01 101000	substracción
Lm	01 001000	multiplicación
Li	01 010000	división
Lw	01 011000	raíz cuadrada

Las operaciones aritméticas utilizan como argumentos los registros R1 y R2 (la raíz cuadrada emplea sólo el registro R1) y almacenan el resultado en el registro R1. Después de transferir un resultado a la memoria, los registros se borran y tienen que cargarse de nuevo.

Un programa para Z3

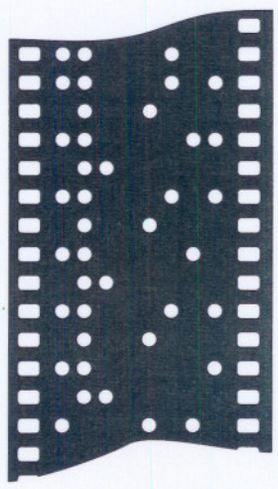
Un problema frecuente en aritmética es el de calcular un polinomio

$$a_4x^3 + a_3x^2 + a_2x^1 + a_1.$$

Con mayor propiedad, esto es, con el método de Horner, podemos expresar el cálculo anterior así:

$$x(a_2 + x(a_3 + xa_4)) + a_1.$$

Suponga que en las direcciones 1, 2, 3 y 4 de memoria hemos almacenado con anterioridad las constantes a_1, a_2, a_3 y a_4. En la dirección 5 hemos almacenado *x*. El programa mostrado abajo realiza el cálculo deseado (por razones mecánicas las ocho perforaciones de la cinta que representan cada instrucción no estaban colocadas en una sola fila. En nuestro diagrama hemos simplificado).



Pr 4	carga a_4 en Registro 1 (R1)
Pr 5	carga *x* en Registro 2 (R2)
Lm	multiplicación, resultado en R1
Pr 3	carga a_3 en R2
Ls_1	suma R1 y R2, resultado en R1
Pr 5	carga *x* en R2
Lm	multiplicación, resultado en R1
Pr 2	carga a_2 en R2
Ls_1	adición, resultado en R1
Pr 5	carga *x* en R2
Lm	multiplicación, resultado en R1
Pr 1	carga a_1 en R2
Ls_1	adición, resultado en R1
Ld	despliega resultado

RAUL ROJAS es catedrático de inteligencia artificial en la Universidad Libre de Berlín. Estudió matemáticas y física en México. Posteriormente obtuvo su doctorado y habilitación en la Universidad Libre. Trabajó durante algunos años en los Laboratorios Nacionales de Computación y Matemáticas de Berlín en un proyecto de ordenadores simbólicos. Desde 1989 investiga en el área de redes neurales. El autor conoció a Konrad Zuse en Berlín y fue en una de esas conversaciones cuando surgió la idea de analizar la estructura y las patentes de la Z1 y Z3.

cutar las cuatro operaciones básicas de la aritmética y extraer la raíz cuadrada de un número. La máquina se controla con una cinta perforada en la que ocho perforaciones codifican cada operación (*véanse las figuras 1d y 2b*). Una vez inicializada la máquina, el programa se ejecuta automáticamente hasta el final. La Z3 no es muy rápida De acuerdo con los parámetros actuales, la Z3 procede con exasperante lentitud: en realizar una multiplicación tarda tres segundos interminables.

Zuse no solamente adoptó el sistema binario, sino que además decidió que sus máquinas operarían con números de coma flotante. Igual que se representan fracciones a través de números decimales, se puede escribir números binarios con cifras antes y después de la coma. Las posiciones antes de la coma indican las potencias 2^0, 2^1, 2^2, etcétera, mientras que las posiciones después del punto señalan las potencias 2^{-1}, 2^{-2}, etc. La representación binaria del número 1/2 es $0{,}1 = 2^{-1}$; la del número 1/4 será $0{,}01 = 2^{-2}$. El número 3/4 tiene la representación binaria fraccionaria 0,11 (*véase el recuadro*).

En el sistema decimal, cuando se opera con números de muy diferentes magnitudes se les escribe como el producto de un número de magnitud restringida, la mantisa, y una potencia de 10. Con esta convención la velocidad de la luz, por ejemplo, es de 3×10^8 metros por segundo y la masa del protón es de $1{,}6724 \times 10^{-24}$ gramos. Con semejante recurso no sólo nos ahorramos muchos ceros, sino que, además, posibilitamos que una máquina que sólo opera con menos de 8 o bien 24 posiciones decimales realice cálculos con ambos números. Para reducir el número de posiciones decimales necesarias y aprovechar esta representación al máximo, hay que "normalizar" los números. Esto significa que antes de la coma decimal ha de haber una sola cifra. Así, podemos expresar el número 8731 mediante la notación $8{,}731 \times 10^3$. En el sistema binario la mantisa está normalizada si representa a un número mayor o igual a 1, aunque menor que 2. El único bit a la izquierda de la coma decimal distinto de cero es el primero. El número 1,01, por ejemplo, está normalizado, mientras que el número 10,1 no lo está. Esta es la convención empleada por Zuse.

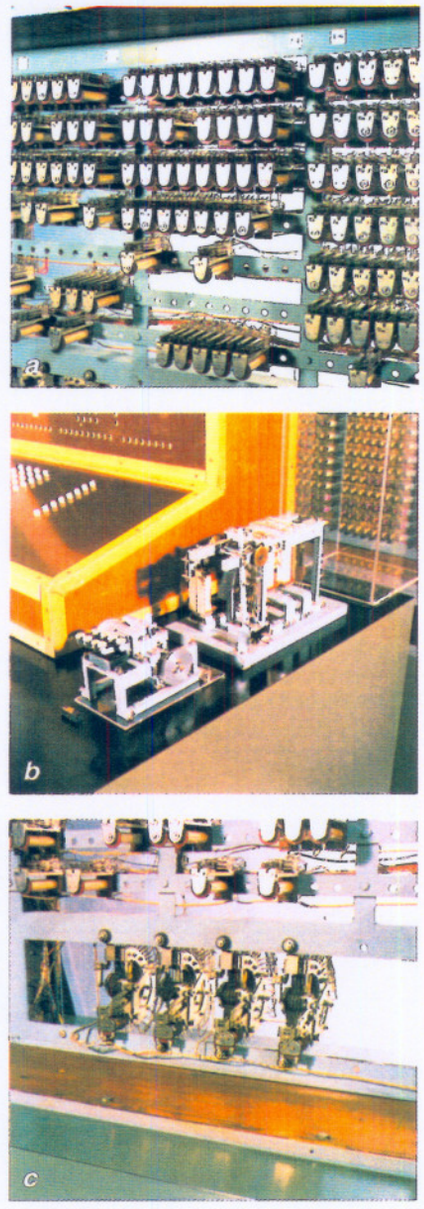
En la memoria de la Z3 se almacenan las mantisas utilizando 14 bits para todas las posiciones de la derecha de la coma. El 1 que precede a la coma no precisa almacenarse, sino que se agrega al transferir un número de la memoria hacia el procesador. El procesador emplea dos bits adicionales en las posiciones más bajas para aumentar la exactitud de los cálculos numéricos. Los ordenadores modernos recurren a una técnica similar para minimizar errores de redondeo.

Los datos para un cálculo se pueden leer a través de un teclado. Se pueden teclear cuatro dígitos para la mantisa (*véase la figura 1b*) y se puede seleccionar un exponente entre –8 y 8 (oprimiendo el botón adecuado en una hilera de botones). El número 4,356 se puede alimentar en la Z3 tecleando la mantisa 4356 y seleccionando el exponente –3. La computadora Z3, al leer este número, lo transforma de la representación decimal en la binaria. Los cálculos posteriores se realizan en binario.

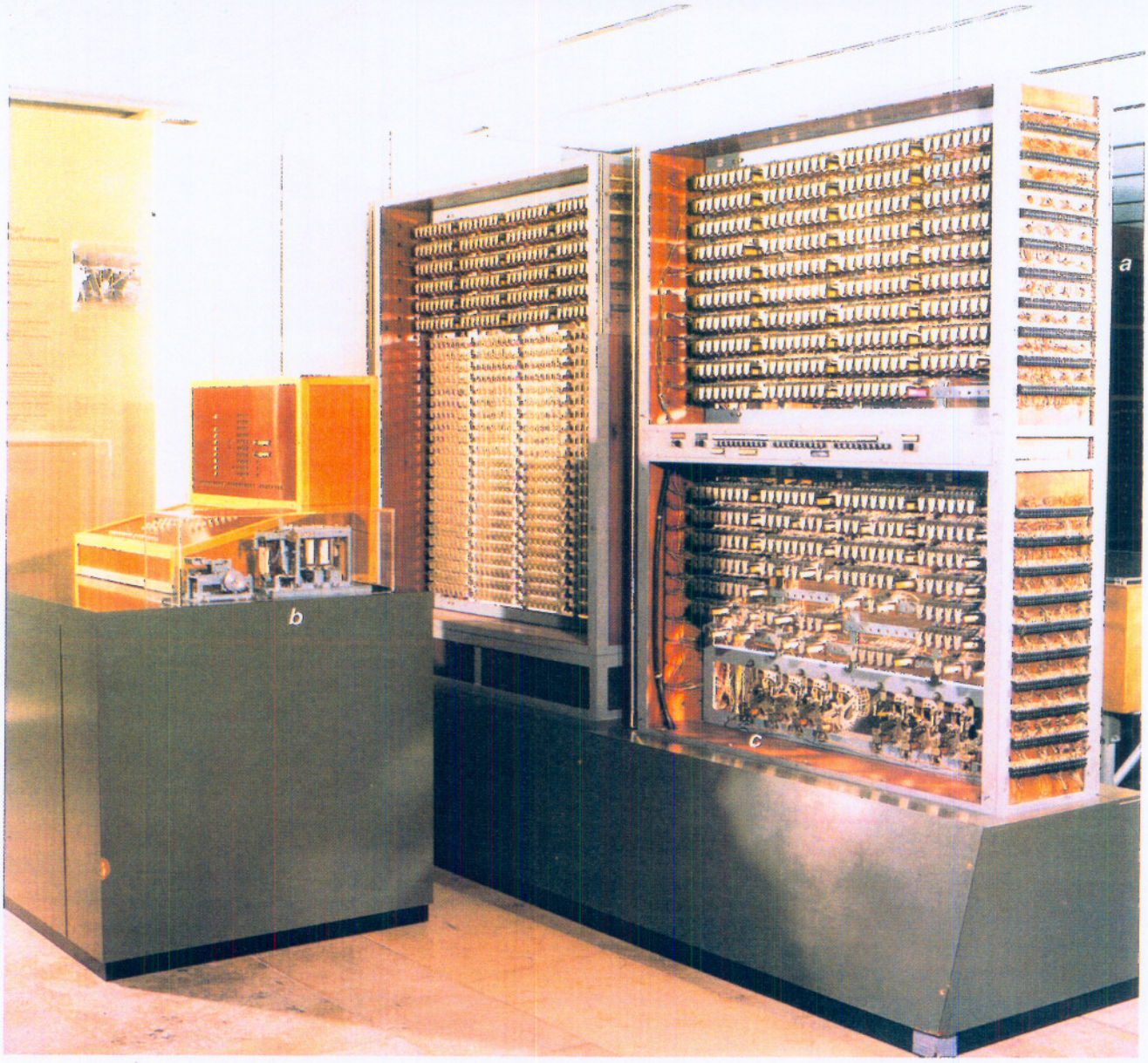
Una vez que deseamos ver el resultado final, la representación binaria interna se convierte en un número decimal con una mantisa entre 0 y 19999; los dígitos correspondientes, así como el exponente, se muestran iluminando una matriz de lamparitas en la consola de control.

La memoria, la unidad aritmética, la unidad de control para las instrucciones y las unidades de entrada y salida de datos numéricos constituyen los principales componentes del tipo de ordenadores construidos por Zuse. La máquina se controla por medio de una cinta perforada de película de 35 mm. Una cabeza lectora lee el código binario expresado por ocho posiciones (perforadas o no) de la cinta. Allí donde hay una perforación, se cierra un contacto eléctrico; donde no la hay, no fluye corriente. Un decodificador inicia la ejecución de la instrucción cuyo código se ha leído. Al final de la ejecución de esta instrucción, la cinta perforada avanza una posición; la instrucción siguiente está lista para su ejecución.

El programador dispone de nueve instrucciones: dos para la entrada y salida de datos, dos para leer datos de la memoria o almacenarlos en ella y cinco para las operaciones aritméticas. La unidad aritmética contiene, desde la perspectiva del programador, dos palabras de memoria que



2. MAQUINA Z3. Se construyó con relés. Tras la consola, el módulo de la izquierda contiene la memoria; el de la derecha, el procesador. Los relés del procesador (*a*) están marcados con los nombres mencionados en el artículo.



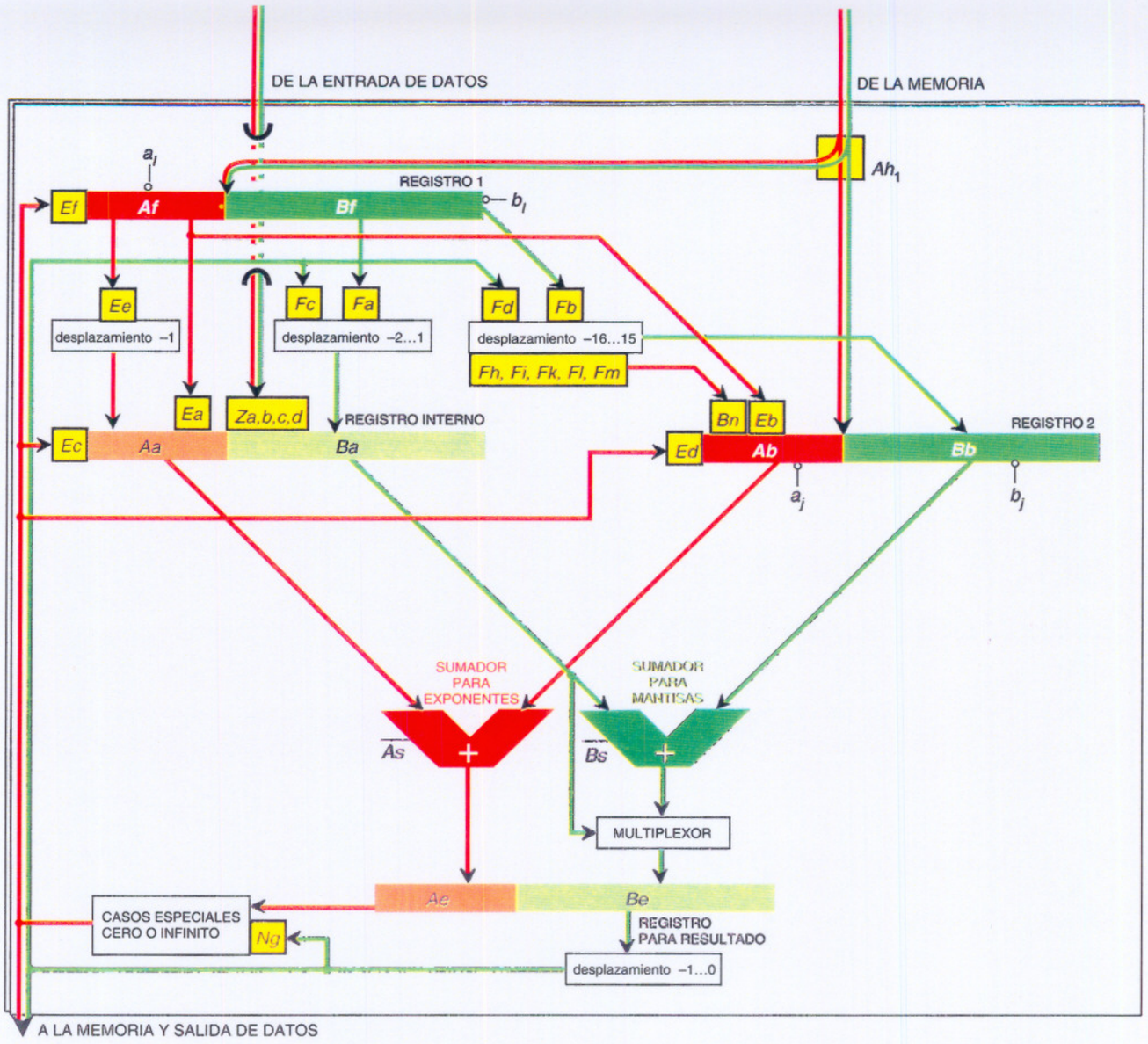
Para ejecutar las operaciones que requieren una secuencia de acciones elementales, Zuse empleó contactos circulares como los utilizados en la telefonía (*c*). En cada ciclo, el brazo móvil avanza un paso y activa el siguiente contacto, lo cual desencadena ciertas operaciones en la máquina. Los programas para Z3 se almacenan en cintas de 35 mm. La cinta se decodifica mediante una unidad de lectura (*b, en la parte posterior*): en donde hay una perforación se cierra un circuito eléctrico. Aquí ofrecemos la reconstrucción de la máquina terminada por Zuse en 1966; se exhibe en el Museo Alemán de la Técnica en Munich. El prototipo original se destruyó en la guerra.

llamamos registros R1 y R2. Todas las operaciones aritméticas se desarrollan sobre estos dos registros.

Cada instrucción de la computadora Z3 se descompone en operaciones elementales, que requieren un ciclo para su ejecución. En la máquina Z1 un ciclo equivale a una rotación de una manivela operada manualmente o de un motor eléctrico (*véase la figura 1b*). La instrucción *Pr* (leer de la memoria) puede realizarse en un solo ciclo; la adición y substracción requieren 3 o 5 ciclos; la multiplicación, división o extracción de la raíz cuadrada necesitan hasta 20 ciclos. Durante la ejecución de estas instrucciones se detiene el avance de la cinta perforada en el segundo ciclo, avance que sólo se reanuda cuando ha terminado la operación aritmética. La instrucción *Ps* (almacenar en la memoria) se puede traslapar con el ultimo ciclo de la operación aritmética previa; no consume, pues, tiempo de máquina.

Cada ciclo (Zuse utilizaba para estos conceptos una terminología propia) se divide en cinco pasos. La instrucción a ejecutar se decodifica en el paso I. Los argumentos para la operación se cargan en los registros durante los pasos IV y V. En el segundo ciclo, en los pasos I, II y III se suman o substraen los exponentes y mantisas (a tenor de la operación de que se trate). En los pasos IV y V se escribe el resultado de la operación en los registros adecuados.

La función de la unidad de control consiste en promover en el procesador las acciones adecuadas en el



momento necesario. En el caso de instrucciones que requieren varios ciclos y se iteran, se emplea un brazo mecánico que se desplaza sobre una carátula con contactos, a la manera de un reloj (*véase la figura 2c*). En cada paso, el brazo mecánico pasa corriente por los contactos a determinados relés y de esa manera se logra que todos operen en concierto. Este tipo de control rotatorio se puede comparar con los microprogramas de los procesadores modernos, que reducen una instrucción compleja a una serie de microinstrucciones, más sencillas.

Para la adición y la substracción existe un circuito de relés que las ejecuta directamente. Antes de comenzar una adición o substracción se comparan internamente los signos de los operandos y se decide si debe ejecutarse la operación contenida en el programa (cuando los signos de los operandos son iguales) o la operación inversa (cuando los signos son distintos).

La multiplicación consta de una serie de adiciones con desplazamiento de la mantisa. El procedimiento no difiere del tradicional papel y lápiz para números decimales. Ahora bien, puesto que se emplea el sistema binario la receta es mucho más sencilla ya que la tabla de multiplicar sólo contiene los números 1 y 0.

El cálculo comienza por igualar el resultado parcial a cero. En cada iteración del algoritmo se examina bit por bit del multiplicador, empezando por el bit más a la derecha. Si el bit examinado en el multiplicador es 1, se suma el multiplicando al resultado parcial; si es 0 no se suma. El nuevo resultado parcial se desplaza un sitio a la derecha y se continúa con la siguiente iteración.

La multiplicación opera con los 14 bits de la mantisa del multiplicando a la derecha de la coma y con el bit a la izquierda. El cálculo consume 16 ciclos (incluido un ciclo para preparar la operación). En tal operación se invierten, como dijimos antes, tres segundos. En cada ciclo de una multiplicación se ejecuta una adición: el resultado parcial se recircula para que sirva de argumento en la iteración siguiente. El multiplicando se mantiene durante toda la operación en el registro R2 y es el segundo argumento en cada adición (*figura 4*).

La división y la extracción de la raíz cuadrada se calculan de manera similar. Si bien en el caso de la división de lo que se trata es de substraer mantisas y desplazar el resultado parcial. El cociente se construye por iteración, bit a bit.

La raíz cuadrada q de un número x se construye también por iteración, de suerte tal que $x/q = q$. Por ello, puede calcularse la extracción de la

3. LA ESTRUCTURA LOGICA de la unidad aritmética de Z3. Los flujos de datos están marcados con colores: rojo para el exponente y verde para la mantisa. Las rectángulos amarillos representan relés que funcionan como válvulas: al abrirse (o cerrarse) permiten (o impiden) el flujo de datos de un registro al otro. El relé *Ah1* tiene como función determinar si los datos que vienen de la memoria han de almacenarse en el registro R1 o en el registro R2 del procesador. Además de estos dos registros, en la figura se muestran otros dos: uno interno y otro para el resultado de las operaciones. Ambos registros quedan fuera de la vista del programador. Los registros R1 y R2 mantienen su contenido indefinidamente hasta que se borran (usando los contactos a_1, b_1, a_j, b_j) o se almacenan nuevos datos en ellos. Los componentes marcados con un símbolo de adición son los sumadores binarios. Se trata de los únicos componentes con los que se computan nuevos resultados. Cuando se activan los contactos *As* o *Bs* las unidades calculan la diferencia y no la suma de sus dos argumentos. En operaciones que requieren varios ciclos, la mantisa y el exponente circulan varias veces iterativamente en el procesador. El exponente tiene su propio circuito (*rojo*) y la mantisa el suyo (*verde*). En cada ciclo se les puede desplazar uno o más puestos hacia la izquierda o la derecha. El exponente, por ejemplo, se puede dividir entre dos, desplazándolo una posición hacia la derecha usando los contactos *Ee*. La mantisa se puede volver a normalizar después de una suma empleando el desplazador (*shifter*) mostrado en la parte inferior de la figura (–1 ... 0). Durante la multiplicación o división se requiere usar el desplazador que traslada la mantisa dos posiciones hacia la izquierda o una hacia la derecha (–2 ... 1). Después de una substracción puede ser necesario normalizar el resultado y requerir quizá de +16 a –15 desplazamientos (16 posiciones hacia la izquierda o 15 hacia la derecha). La corrección para el exponente se transmite a través de los relés *Fh, Fi, Fk, Fl, Fm*. Antes de que un resultado continúe circulando en el procesador, se verifica si no se trata de un caso especial (cero o infinito) y, si es así, se activan circuitos especiales que producen la codificación necesaria. Los datos decimales introducidos a través del teclado se transmiten al registro *Ba* vía las válvulas *Za, Zb, Zc* y *Zd*.

raíz cuadrada siguiendo un algoritmo muy similar al de la división y haciendo uso de los mismos circuitos.

Algunas operaciones se complican al adoptar la notación de coma flotante. Ocurre así con la adición y substracción. Antes de sumar dos números de coma flotante hay que hacer coincidir la posición de la coma: los exponentes tienen que ser iguales. Pero si los exponentes son distintos, la mantisa del número con el exponente menor se desplaza hacia la derecha (incrementando cada vez el exponente) tantos sitios cuantos se precise para que el exponente menor se iguale con el exponente mayor.

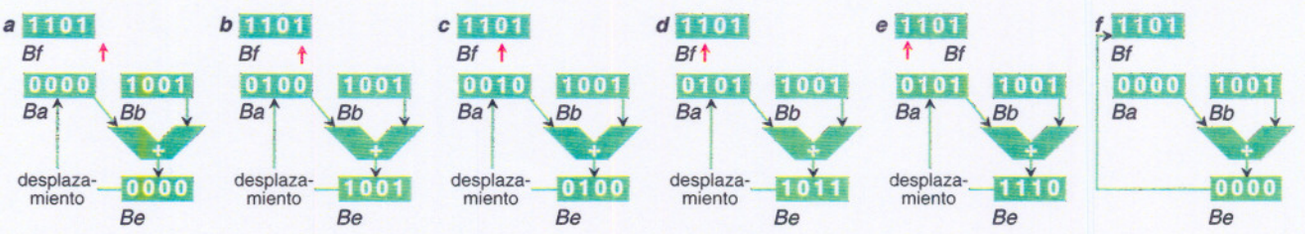
En el caso de la multiplicación, el procedimiento resulta más sencillo. Basta con sumar los exponentes y multiplicar las mantisas. En una división se substraen los exponentes y se dividen las mantisas.

Que la operación haya terminado no garantiza que el resultado esté normalizado. El producto de dos mantisas con valores entre 1 y 2 puede ser mayor que 2 (y eso significa que el segundo bit a la izquierda de la coma es 1). También puede suceder que al substraer dos números casi iguales aparezcan en la mantisa del resultado ceros en los primeros bits de izquierda a derecha. En ambos casos se debe desplazar la mantisa del resultado hasta que en el primer bit antes de la coma (y en ninguno más arriba) haya un 1. De acuerdo con la dirección del desplazamiento usado para normalizar la mantisa, habrá que sustraer, o sumar, al exponente el número de sitios corridos.

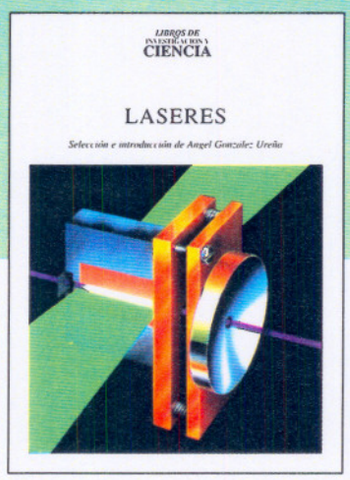
De lo anterior se infiere, pues, que la operación fundamental realizada en la máquina Z3 es la adición de exponentes y mantisas. La computadora necesita un ciclo para esta operación, lo mismo que los microprocesadores modernos. En el caso de operaciones cuyo resultado se obtiene por iteración, el resultado parcial atraviesa varios ciclos de procesador.

Por la época en que Zuse creaba sus prototipos, en Estados Unidos otros inventores desarrollaban proyectos similares. Entre 1938 y 1942 John Atanasoff construyó en la Universidad estatal de Iowa la máquina ABC (de Atanasoff-Berry Computer). De 1939 a 1944 Howard Aiken dirigió en la Universidad de Harvard el desarrollo de la máquina Mark I, mientras que en la Universidad de Pennsylvania J. Prespert Ecker y John W. Mauchly construían entre 1943 y 1945 el ENIAC. Resulta muy ilustrativo comparar las características de estos cuatro instrumentos de cómputo.

Un aspecto fundamental de los ordenadores modernos es la separación de memoria y procesador. Ni en Mark I ni en el ENIAC se llegó a materializar esa partición. En ambos computadores las celdas de memoria se hallaban directamente acopladas a circuitos para cálculos numéricos. Por esa razón, y en proporción a su li-



4. MULTIPLICACION DE DOS MANTISAS en el procesador de Z3. Para simplificar el diagrama utilizamos registros con cuatro bits (en vez de 15). Al principio (*a*) el multiplicando 1,001 (= 1,125 en decimal) se encuentra en el registro *Bb* (que representa la mantisa del registro R2) y el multiplicador 1,101 (= 1,625 en decimal) en el registro *Bf* (mantisa del registro R1). El registro interno *Ba* para los resultados intermedios está inicializado a 0. En cada ciclo (*b* hasta *e*) se suma el multiplicando con el resultado parcial, siempre y cuando el bit correspondiente del multiplicador (*flecha roja*) sea 1. El nuevo resultado parcial se desplaza una posición hacia la derecha y se almacena en *Ba*. La última posición binaria se pierde en esta operación (error de redondeo). Cada uno de los diagramas (*b*) a (*e*) muestra el estado del procesador antes de que se ejecute la adición. En el último ciclo (*f*) se escribe el resultado final 1,110 (= 1,75 decimal), ahora sin desplazamiento, en *Bf*. El resultado, aunque parece falso, es igual al resultado decimal correcto 1,828125 cuando se redondea su representación binaria utilizando cuatro bits para la mantisa.



- EL LASER C^3, W. T. Tsang
- LASERES DE RAYOS X BLANDOS, Dennis L. Matthews y Mordecai D. Rosen
- LASERES DE ELECTRONES LIBRES, Henry P. Freund y Robert K. Parker
- MICROLASERES, Jack L. Jewell, James P. Harbison y Axel Scherer
- APLICACIONES DEL LASER EN LA INDUSTRIA, Aldo V. La Rocca
- CIRUGIA CON LASER, Michael W. Berns
- INTERACCION DE LA RADIACION LASER CON LOS MATERIALES, Carmen Ortiz
- AVANCES EN LA FUSION POR LASER, R. Stephen Craxton, Robert L. McCrory y John M. Soures
- DETECCION DE ATOMOS Y MOLECULAS CON LASERES, Vladilen S. Letokhov
- LA FISICA DE SUPERFICIES, Rodolfo Miranda
- IMPLANTACION IONICA DE SUPERFICIES, S. Thomas Picraux y Paul S. Peercy
- REACCIONES QUIMICAS Y HACES MOLECULARES, Angel González Ureña
- LA FORMACION DE LAS MOLECULAS, Ahmed H. Zewail
- ESPECTROSCOPIA DE GASES SOBREENFRIADOS, Donald H. Levy

mitada capacidad de cálculo, resultaban unas máquinas en exceso complejas. La verdad es que sólo Z3 y ABC eran computadores exclusivamente binarios. En Mark I y ENIAC se utilizaban internamente números decimales, aun cuando los componentes lógicos eran binarios. Pero ninguna de las cuatro máquinas proporcionaba al programador instrucciones para modificar el flujo del programa de acuerdo con el valor de una variable determinada (bifurcación condicional). En computadoras universales estas instrucciones de transferencia son imprescindibles; si así no fuera, se haría imposible modificar el curso del cálculo en coherencia con los resultados parciales. El ENIAC podía realizar cierto tipo de bucles (secuencias repetitivas) y transferir el control al final de su ejecución; para eso disponía de sólo una unidad de control. Zuse introdujo instrucciones de transferencia en otros de sus diseños de años posteriores.

Importa señalar que, de los cuatro computadores, únicamente Z3 utilizaba números de coma flotante. Z3 y Mark I eran dispositivos electromecánicos; ABC y ENIAC se basaban ya en tubos de vacío. ABC se proyectó para operar con vectores y, como tal, se le considera un precursor de los procesadores vectoriales al uso. La arquitectura del ENIAC encarna lo que llamaríamos "arquitectura de flujo de datos". No se escribía un programa, sino que se modificaban las conexiones de una unidad a otra: los datos fluían a través de la máquina y se transformaban en cada unidad (con adición, substracción, etc.). ABC y ENIAC carecían, por tanto, de programación lógica ("software"). El computador ABC sólo podía ejecutar el programa materializado en sus circuitos: el algoritmo de Gauss para la solución de sistemas de ecuaciones lineales. Z3 y Mark I, por contra, leían sus programas de cinta perforada.

Zuse y Atanasoff construyeron sus respectivos ordenadores en solitario. Lo que no ocurrió con Mark I ni con ENIAC, en cuya concepción y realización colaboraron grupos importantes. Zuse y Atanasoff tuvieron, pues, que refinar al máximo sus proyectos para no gastar ni un tornillo de más. El diseño arquitectónico entrañado en Z3 posee, sin embargo, una mayor madurez y señala el camino que se seguirá en el futuro, lo que no se percibe en ABC, un modelo de máquina más limitado.

Sabido es que la victoria tiene muchos padres y la derrota es huérfana. Aplicado ese aforismo a la historia del ordenador, no podemos hablar de un instante decisivo ni de un inventor clave. Lo que se dio fue un largo proceso de desarrollo que culminó en las máquinas de cómputo universal que usamos hoy en día. Muchas ideas, grandes y pequeñas, hubieron de entremezclarse con los conceptos básicos de Turing. Ni deben olvidarse los precursores, con Charles Babbage (1791-1871) al frente, quien concibió un autómata programable al que llamó Máquina Analítica. John von Neumann (1903-1957) influyó de manera importante en el diseño de los ordenadores construidos después de la guerra. Aunque de sus manos no saliera ningún ordenador, se mantuvo siempre en la primera línea, aportando ideas sobre teoría de computabilidad y ayuda a otros.

Hace algunos años dicté una conferencia sobre algunos de los temas abordados en este artículo. En la última diapositiva había tres nombres: Babbage, Zuse y von Neumann, por ese orden. Cuando en 1993 le enseñé la filmina a Zuse, confiado en que se alegraría de ver su nombre en tan ilustre compañía, me espetó: "¿Qué hace ahí von Neumann?

Así de seguro en su valía era el ilustre inventor. Nadie puede objetar, empero, que su puesto en la historia de la técnica se lo ganó a pulso. A él le debemos que en 1936 terminara la memoria mecánica de 1936, el autómata que anunciaba desde un pequeño apartamento de Berlín el inicio de una nueva era, la era de la computación.

BIBLIOGRAFIA COMPLEMENTARIA

DER COMPUTER–MEIN LEBENSWERK. K. Zuse. Springer-Verlag, Berlín, 1984.

"WHO INVENTED THE COMPUTER? THE DEBATE FROM THE VIEWPOINT OF COMPUTER ARCHITECTURE". R. Rojas en *Fifty Years Mathematics of Computation, Proceedings of Symposia in Applied Mathematics*, AMS, coordinado por W. Gautschi, páginas 361-366, 1993.

DIE RECHENMASCHINEN VON KONRAD ZUSE. Dirigido por R. Rojas. Springer-Verlag, Berlín, 1996.

KONRAD ZUSE'S LEGACY: THE ARCHITECTURE OF THE Z1 AND Z3. R. Rojas, en *IEEE Annals of the History of Computing*, vol. 19, n.º 2, páginas 5-16, 1997.