Synchronization of a Magnetic Computer

J. KIELSOHN

G. SMOLIAR

IN COMPUTER DESIGN the problem of communication between the central computer and the outside world is always present. This paper describes how this problem was solved for the Cambridge computer with a new type of shift register employing the Ferractor*, the magnetic core developed by Remington Rand Univac. The computer and the magnetic amplifiers which it uses were described in some detail in other papers. The shift register itself has many unique features, however, and the design and application merit a separate discussion.

The Cambridge computer is a magnetic device in the sense that all of the normal amplifying functions are performed with Ferractors. Germanium diodes are used for the logical operations of gating and buffering. Pulse shaping and delay are accomplished by the magnetic amplifiers, and therefore separate components are not needed to perform these functions.

The synchronizer of a computer must act as a link in both directions between the essentially uniform pulses circulating in the computer and the input and output pulses occurring at random. The Cambridge computer, although relatively restricted in input-output functions, still has many external communication paths. There are data links for real-time operation, an in-and-out paper-tape punch and reader, and a modified typewriter for input and output. Consideration of the typewriter only, however, will involve all of the fundamental principles of synchronizer operation.

Transfer of data from the typewriter is made through a shift register, R₀, in which the ten characters of a computer word are assembled. The digits coming from the typewriter occur at random and in reverse order with respect to the sequence in which they are carried in the computer. In other words, an operator types the most-significant digit first, while the word is carried in the computer with the least-significant digit first, for the usual reason that the result of a carry may be recorded after the carry.

From the collection of the Computer History Museum (www.computerhistory.org)
sign of a word is stored separately during computing, and, even though it is part of the word-pulse sequence within the memory, it presents no special synchronizing problems and it will not be discussed here.

Before discussion of the actual circuit of the shift register used for data transfer, the magnetic amplifier from which it is derived will be described. In the simplest form, this amplifier consists of two windings on a toroid of square hysteresis-loop magnetic material. A pulse applied to the input will bring the core from the normal state of plus $B_r$ to the opposite point on the hysteresis loop, minus $B_r$; see Fig. 1. The core will then present a high impedance to a power pulse, and the power pulse will return the core to plus $B_r$ without producing appreciable output. If there is no input pulse, the core stays at plus $B_r$ for the entire input period, and the power pulse, encountering a low impedance, produces an output pulse; see Fig. 2.

Since the power required to flip the core is much less than the power that can be sent through the output winding, the device is an amplifier, and it is easy to see that the core also reshapes and delays the pulses as in Fig. 2. This circuit, with only minor modifications, is the one that is most frequently used in the Cambridge computer. It complements the information; that is, it substitutes pulses for no pulses and vice versa. The circuit is modified to make a true amplifier by the addition of a bias winding which carries direct current and which is phased opposite to the input winding; see Fig. 3. In the amplifier, an input pulse brings the core to plus $B_r$, so that the power pulse finds it in the low-impedance state, while during the absence of an input pulse the bias current drives the core to minus $B_r$, with the result that the succeeding power pulse will produce no output.

Recirculating registers may be made by cascading strings of these amplifiers and connecting the output of the string back to the input. However, for the assembly and transfer of asynchronous information a true shift register is preferred; that is, a component which can hold the information in an essentially static form or circulate it, either at computer rate or at one pulse space at a time. The Cambridge computer shift register is based on the noninverting amplifier described above. It has two input-circuit differences and one difference in operation timing. Fig. 4 shows the shift-
register circuitry. Note that there is an additional input winding for each amplifier, and a second core (besides the amplifier) with two windings for each bit of information.

While the shift register operates with the same timing as the rest of the computer, the action is in three steps instead of the two steps, input and power, of the simple amplifiers. Information enters the shift register through the lower Ferractor, shown in Fig. 4, which is called the blocking core. A pulse applied at the external input terminal makes the blocking core a low impedance. One half-cycle later an interrogating pulse in the hold winding tries to send current through both cores. If the blocking core has been brought to low impedance by the input, the hold pulse brings the amplifying or output core to plus $B_T$, so that on the third half-cycle the power pulse will produce an output. When the shift register is used for static information storage, a series of hold pulses causes the information to circulate back and forth between the output and the blocking cores. The output of the upper core is connected through a diode to the input of the lower core; see Fig. 4. If no shift or hold pulses are applied for 1 cycle, the register is cleared, since the bias resets the output core and the negative excursion of power pulse 2 resets the blocking core. Shifting is accomplished by reading into an output core from the preceding blocking core, rather than from the one shown directly below the output core in Fig. 4. In this way, a series of shift pulses will move the information through the register at computer rate. Interconnection of shift-register stages is accomplished by joining the internal out to the internal in terminal of the stage immediately to the right. In these circumstances, information would move from left to right. Obviously, either direction could have been chosen, or, if both were needed, one more winding on each output core and one more diode at the CR13-CRI junction would suffice. In operation both left and right shifts are needed, but, since plenty of time is available, the left shift is accomplished by moving the information ten places to the right in the 11-bit loop. Information can enter the computer in parallel through all of the external input terminals, or serially through any of the terminals with a series of shifts. It can be read out in either form as well.

The operator register of the Cambridge computer, which is used for most input-output operations, consists of 4 loops of 11 bits each, each loop containing one of the 4 bits of a character. All entries in the register are made at the most-significant digit position. Information from input devices is presented to the computer in the form of five bits (the fifth being a parity bit for checking purposes) and a sprocket. Since the sprocket appears at an arbitrary time and must trigger logical operations in the computer, it must be shaped to a computer-size pulse and properly timed. The sprocket pulse is shaped through a resistance-capacitance circuit.
network so that it is longer than 3 computer-pulse times but shorter than 2 word times or 24 computer-pulse times. This pulse then sets flip-flop A (Fig. 5), and shapes the sprocket to computer-pulse size. To time the pulse is the function of flip-flop B; see Fig. 5. At the proper time the state of flip-flop A is sampled. The flip-flop may be set, partially set, or reset. The result of this sampling is placed in flip-flop B. If flip-flop A was fully set, flip-flop B will be fully set, but if flip-flop A was in the process of being set, a partial set will be passed to flip-flop B. (It is the possibility of a partial set in flip-flop A that makes flip-flop B necessary.) The partial set is allowed to recirculate in flip-flop B long enough to cause the partial set to either disappear or build up to full size. The result is then sampled and either a full-sized sprocket or no pulse is obtained. If no pulse is received, a second sampling of flip-flop A is made which will always result in a full set of flip-flop B, and (at the end of the waiting time) a full-sized sprocket. Generation of the sprocket clears flip-flops A and B, allows the four information bits of the input data to be placed in the most-significant position of R0, and starts R0 shifting to the right ten places. Since the register is 11 digits long, the information is now in the least-significant position. This arrangement ensures that the computer will operate on the least-significant digit first. The computer is now ready to accept another input character.

This operation can be terminated either by a special fill character or by a counter which keeps track of the number of characters in the word. Then, by means of a full word of shift pulses, the information, which is now in computer sequence, goes out of the register at computer rate, and the transfer operation is complete.

Like most of the computer components, the shift registers are made up of standard printed-wiring cards, which are interconnected by means of the backboard wiring of the machine. Fig. 6 shows the shift register on a single card which holds four cores and the associated circuitry; that is, 2 bits of shift-register storage.

The versatility of the shift register makes testing of it in the computer very simple. In normal sequence of testing, operation of the typewriter is checked out and then used to generate the characters to fill the shift register. Operation of the register is observed when it is holding information in static form and when it is continuously shifting. A check is made to see that the characters are correct and that they can be cleared out by dropping the hold and shift lines.

Maintenance experience on these components has been very satisfactory to date. Another example of the excellent reliability of Ferractor magnetic-core amplifiers.

---

**TX-O, A Transistor Computer with a 256 by 256 Memory**

**J. L. MITCHELL**  
**K. H. OLSEN**

**Synopsis:** TX-O is a high-speed digital computer which was built at Lincoln Laboratory to demonstrate and operationally test 5-megahertz transistor circuitry and a 65,536-word magnetic-core memory. The word length is 19 bits; 1 bit is a parity check bit for memory, 16 bits are assigned to memory addressing, and the remaining bits are used to select among three memory-reference instructions and one microprogramming instruction. The logic is performed by standardized packages using surface barrier transistors. Fig. 1 shows TX-O with the arithmetic element just beyond the console and the memory on the far left. Part I of this paper covers the TX-O memory, and Part II the TX-O circuitry.

**Part I, The TX-O Memory**

The TX-O MEMORY, Fig. 2, is a high-speed, random-access, coincident-current magnetic-core unit with a storage capacity of 65,536 19-bit words. The bits in the word are read out in parallel, and the cycle time is 7.0 μsec (microseconds). (Cycle time is defined as the time between successive read operations.) Two 256-position magnetic-core switches are used to supply the read and write current pulses to selection lines. The memory system contains 425 dual triodes and 625 transistors. It is interesting to note that the presently available 4,000-register magnetic-core memories use almost as many active elements as are used in this 65,000-register memory. The memory was designed both electrically and mechanically so that the word length can be expanded to 37 bits. A block diagram of the memory system and the timing diagram is shown in Fig. 3. The basic operation of this type of memory system has been adequately described in the literature and will not be repeated here.

**Memory Array**

The memory array contains 115/4 million ferrite cores which were manufactured at the Lincoln Laboratory. The outside diameter of the core is 80 mils, the inside diameter 50 mils, and the height 22 mils. When driven with an 820-milliamper single pulse, the cores switch in 1 μsec and give a peak output voltage of 100 millivolts. The cores used in this memory have a somewhat greater signal-to-noise ratio than available commercial cores. The cores are wired into 64 by 64 subassemblies, each subassembly being a complete operating memory plane with its own sense and digit winding. The same winding configurations are used in the 64 by 64 subassemblies as were used in the previous memories built at the Lincoln Laboratory. Sixteen 64 by 64 subassemblies are assembled in a square array and connected together to form each 256 by 256 plane. The choice of a 64 by 64 subassembly size was a compromise between the number of soldered connections in the 256 by 256 plane and the ease of construction and test of the subassemblies.

The digit-plane winding in each 256 by 256 plane is divided into quarters, each quarter being made up of the digit winding of four subassemblies connected in series as shown in Fig. 4. Each quarter looks like a delay line with a characteristic impedance of 150 ohms.

---

From the collection of the Computer History Museum (www.computerhistory.org)