B

BASIC

All home computer owners should be familiar with this term, and most will know that it supposedly stands for Beginners All-purpose Symbolic Instruction Code. Its origins are less well known. BASIC was developed at Dartmouth College, USA, not as a language for developing software but for teaching programming.

It was really a derivation from FORTRAN, the most popular language among scientists, engineers and academics at the time. BASIC cut down on the complex syntax of FORTRAN (and on its range of functions). In particular, it replaced the hard-to-use WRITE and FORMAT statements with the simple PRINT command.

But the major breakthrough was that BASIC was designed to be interactive: typed in and operated by someone at a terminal, rather than as a stack of pre-punched cards. This is why all BASIC lines have line numbers, so that they can be referred to and edited. Editing a FORTRAN program had meant finding and altering the appropriate cards.

Flip-Flop
A bistable or flip-flop can store or 'remember' a single bit. Combinations of these can be used for storing whole binary numbers

BCD

Binary-coded decimal (or *BCD*) is a method of storing decimal numbers in binary form (for example, in RAM or on disk). Most home computers, however, favour the floating point format in preference to BCD, because it is more efficient in terms of memory usage. Using floating point, a number is converted into one long binary number and then normalised (the radix point — see page 148 — is shifted and the number separated into a mantissa and an exponent). It is then stored in a pre-determined number of bytes, which on home computers is commonly five.

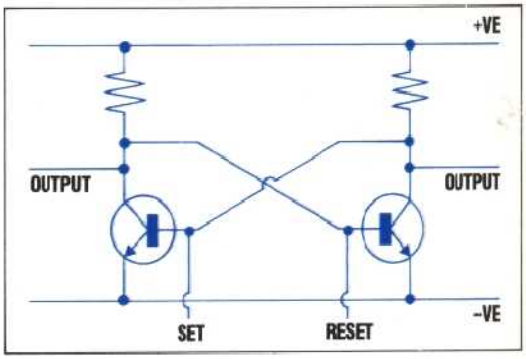
With BCD, each digit in the original decimal number is converted into a four-bit binary number (half a byte), so the number of bytes occupied will correspond to half the number of decimal digits. The computer performs all arithmetic on BCD numbers in a way that is very similar to the way that we perform long multiplication or division (working on each digit of the number separately), whereas a floating point subroutine would treat the number as a whole.

The major advantage of BCD is that it doesn't produce the kind of rounding off errors that we often associate with computers and pocket calculators. This can be particularly important in major banking and financial applications.

BENCHMARK

In the early days of microcomputing (when the PET, Apple II and Tandy TRS-80 were the predominant machines) a set of *benchmarks* was developed to determine the relative speed and efficiency of the BASIC interpreters. These consisted of 10 simple routines that tested different aspects of the BASIC (the speed of execution of loops, floating point arithmetic, trigonometric functions, etc.). You can still find the results of these tests printed in magazines that undertake technical reviews, where they take the form: 'BM1 – 10.2 seconds, BM2 – 3.87 seconds...' and so on.

Attempts to introduce a parallel system for modern business microcomputers have met with little success. This is largely because the throughput of a business system is heavily dependent on the way that the applications software is written. A microcomputer that can execute the XYZ accounting package faster than any other machine may well be the slowest on the ABC database. The eight-bit Osborne 1, for example, is not renowned as being a fast machine, yet it is liked by many journalists because it can execute the Wordstar word processing program faster than most of the new 16-bit computers.



LIZ DIXON

BISTABLE

The *bistable* is one of the simplest of electronic circuits — you can construct one from just two transistors and a handful of resistors — yet the microcomputer owes its very existence to this invention. As the name suggests, a bistable circuit is one that has two stable states, usually indicated by an output line that is 'high' (around five volts) or 'low' (zero volts).

The bistable circuit can therefore be thought of as a single memory bit, capable of storing a '1' or a '0'. The first solid-state semiconductor memories consisted of banks of transistors configured to form an array of bistables. Static RAM chips, which are still found in quite a few microcomputers, are nothing more than miniaturised arrays of bistables. However, modern designs tend to favour dynamic RAMs, which store the information in the form of electrical charges applied to tiny capacitors. As these charges tend to leak away, they have to be constantly refreshed by a special electronic circuit built into the chips. However, dynamic RAMs are faster, and consume less current, than their predecessors.

Bistables can still be found in the discrete logic section of your computer's printed circuit board. They are colloquially known as *flip-flops*, because of their ability to alternate between two states. Flip-flops differ in the way that the state is changed: some have one input line and change state whenever a pulse is applied. But the most common form is the *J-K flip-flop*, which has two inputs (labelled J and K). Applying a voltage to J will initiate one state, while a voltage applied to K will give the other.