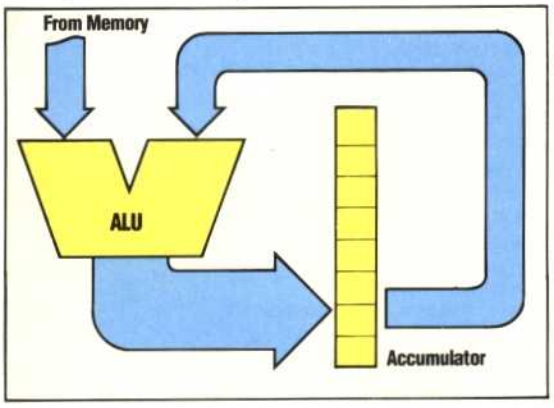
LOGICAL END

The central processing unit (CPU) controls the transfer of data around the computer, oversees the execution of program instructions, and carries out arithmetic and logical calculations. In this final instalment of the course, we look at the logic of the CPU and, in particular, the function of the arithmetic and logic unit (ALU).

The functions that the ALU performs fall into two categories: arithmetical functions — addition, subtraction and incrementation (adding one to a number) — and three logical functions — Exclusive OR, Inclusive OR and AND.

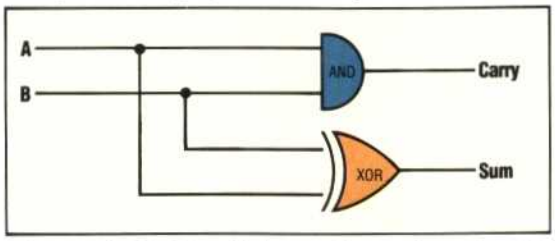
Some of these functions, such as addition, require two operands (i.e. numbers on which the operation is carried out). Some, such as incrementation, require only one operand. In the latter case, the required operand is taken from a special register within the CPU called the accumulator. Where two operands are required, the second is fetched from main memory. The two numbers then progress through the circuitry of the ALU and the selected operation is carried out. The result of the operation is finally placed back in the accumulator.

The following diagram shows the flow of data through the ALU chip:

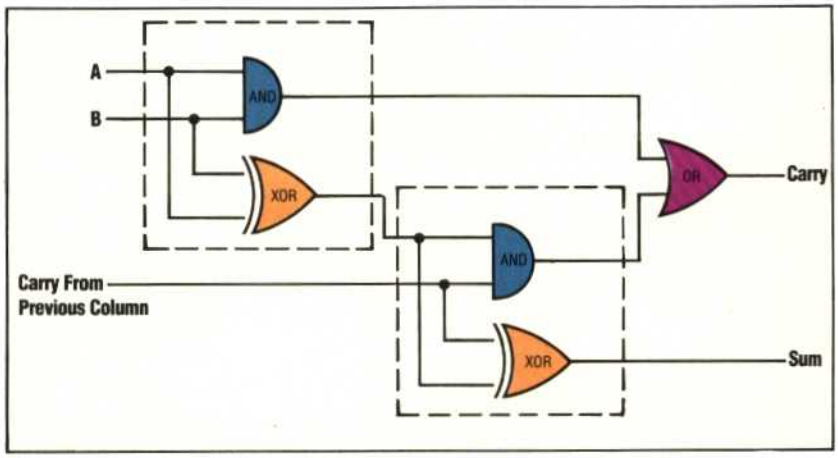


The numbers in the accumulator and memory have an eight-bit word length. The bits that make up the numbers travel through the ALU in parallel and the operation is carried out on all eight bits simultaneously. In order to describe the circuitry of the ALU, let us take just one of the eight bits and design a circuit that will allow us to carry out all six of the different functions. We shall call the bit from the first operand A and that from the second operand B.

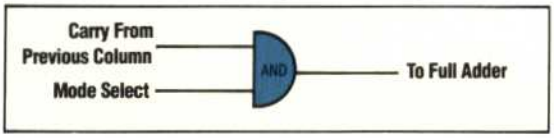
The basis for the one bit stage of the ALU we are considering is the full adder. The full adder circuit we designed in a previous instalment (see page 46) used two half adder circuits, which consisted of AND, OR and NOT gates. These half adders, however, can make use of an Exclusive OR gate to simplify the circuitry:



The two half adders link together to form the full adder circuit in this way:



By adding on small circuits to the full adder it is possible to show how the adder's gates can be adapted to perform the other functions of the ALU. We shall set up a series of control signals to do this, the first of which is known as the *mode select* signal. The 'carry from previous column' input is used during arithmetic operations but is not required during logical operations. The mode select signal switches the carry input on or off. This is easily achieved by using an AND gate.



For arithmetic operations where the carry input is required, the mode select signal should be set to one, allowing the carry input to pass through the AND gate. When the carry input is not required then the mode select signal should be set to zero. In a similar way, we can add AND gates to the two other inputs. This allows us to select either bit A, bit B or both bits together.

During the process of subtraction by two's complement additions (see page 278), the two's complement of the number to be subtracted must be calculated. This requires the changing of all ones to zeros, and vice versa. If our adder circuit is to be used for subtraction then we must add some circuitry that will allow us to select the negation of bit B. This can be done by feeding the input B through a NOT gate, and the select signal can be incorporated by means of another AND gate. The