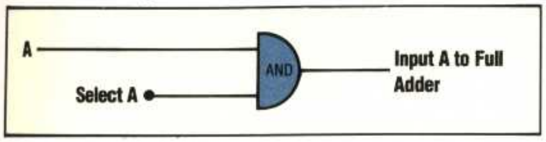
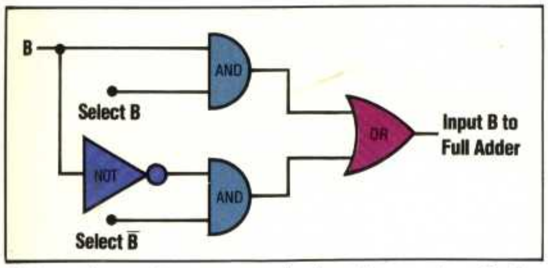
final circuitry for the inputs A and B together with the control signal is as follows:



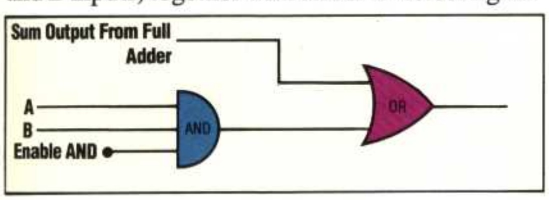


Using these four control signals, each of the arithmetic functions can now be accomplished. This table shows the required combinations:

Function	Mode Select	Select A	Select B	Select B̄
Addition	1	1	1	0
Subtraction	1	1	0	1
Increment A (Set 1st carry input to 1)	1	1	0	0
Increment B	1	0	1	0

As the logical functions do not require the 'carry from previous column' input, we can set the mode select signal to zero for all logical operations. This means that the logical XOR function can be achieved at the sum output by setting Select A and Select B HI, and setting the mode select signal LO.

The AND function cannot be taken directly from the existing circuitry and requires separate A and B inputs, together with an AND select signal.



Finally, the OR function can be created by combining the XOR and AND outputs through an OR gate. The following truth table demonstrates this:

A	B	Output	Comments
0	0	0	
0	1	1	XOR Function
1	0	1	
1	1	1	AND Function

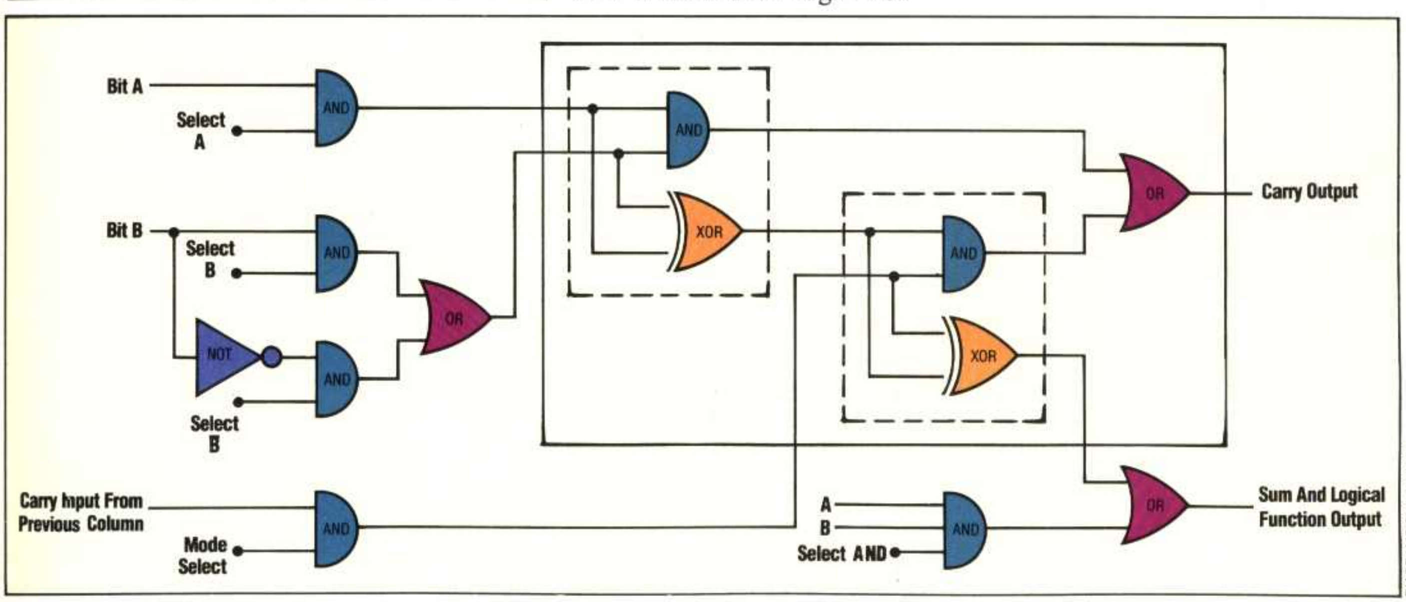
The following table shows how the logical functions can be produced using different combinations of the control signals:

Function	Mode Select	Select A	Select B	Select B̄	Select AND
XOR	0	1	1	0	0
AND	0	0	0	0	1
OR	0	1	1	0	1

The final diagram below shows a one-bit stage of an ALU circuit, illustrating the full adder circuit and all the additional circuits for the control signals. The full circuit would incorporate eight such circuits in parallel. The carry output from the eighth column is used as the carry flag for the processor status register.

This article concludes our series on logic. We started the course by dealing with such abstract logical concepts as Boolean algebra and Venn diagrams, and then considered simple logic circuits and the results they give. The last articles in the series investigated more complex circuits at a level close to that used inside a computer.

Finally, we have shown how it is possible to combine several circuits in a way that allows a microprocessor to perform all the arithmetic and logical operations it needs to do. Each operation can be performed by putting the right patterns of ones and zeros on the command lines of the ALU. These patterns are in fact machine code instructions in their true binary form. Thus we have studied the logic of the hardware inside computers up to the point where software takes over as the controlling factor.



DIAGRAMS BY LIZ DIXON