

Lab02

Carry Select Adder

A carry select adder is an arithmetic combinational logic circuit which adds two N-bit binary numbers and outputs their N-bit binary sum and a 1-bit carry. This is no different from a ripple carry adder in function, but in design the carry select adder does not propagate the carry through as many full adders as the ripple carry adder does. This means that the time to add two numbers should be shorter. It seems this optimisation was used by Charles Babbage in his design for the Analytical Engine.

The idea behind a N-bit carry select adder is to avoid propagating the the carry from bit to bit in sequence. If we have two adders in parallel: one with a carry input of 0, the other with a carry input of 1, then we could use the actual carry input generated to select between the outputs of the two parallel adders. This means all adders could be performing their calculations in parallel. Having two adders for each result bit is quite wasteful so we could configure the N-bit adder to use $2*N/M-1$ M-bit ripple carry adders in parallel. Note that the adder for the least significant bits will always have a carry input of 0 so no parallel addition is needed in this case.

Example

As an example, an 8-bit carry select adder could comprise three 4-bit ripple carry adders: one would calculate the sum and carry for the low nibble sum (bits 0 to 3), and the other two would calculate the high nibble sum and carry (bits 4 to 7). All adders would calculate in parallel. We could then use the low nibble carry output as a selector for a multiplexer that would choose the correct results from the high nibble sums and carries.

