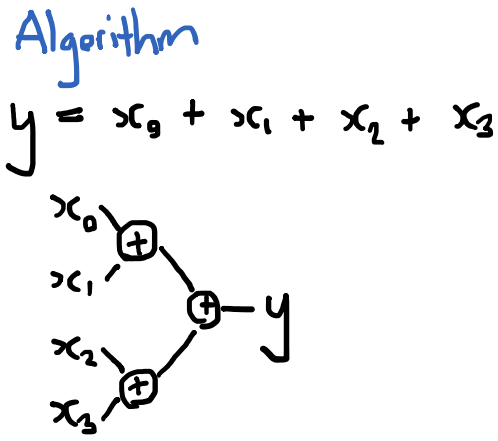
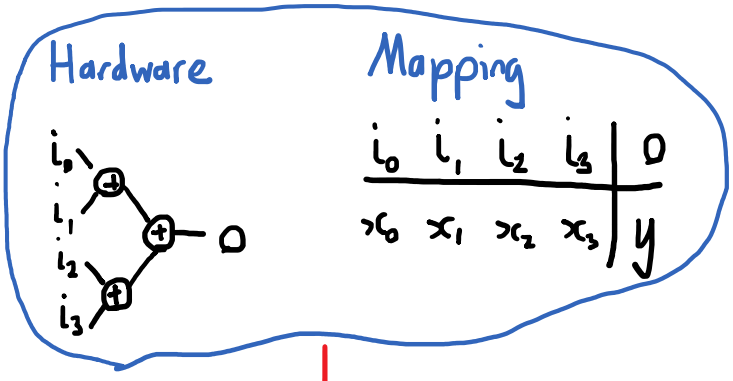
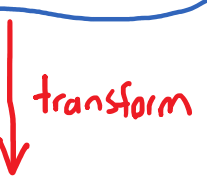
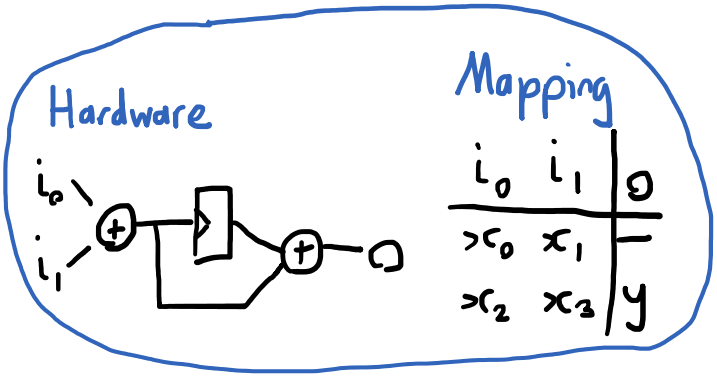
Adding 4 Numbers

Monday, May 9, 2022 2:52 PM









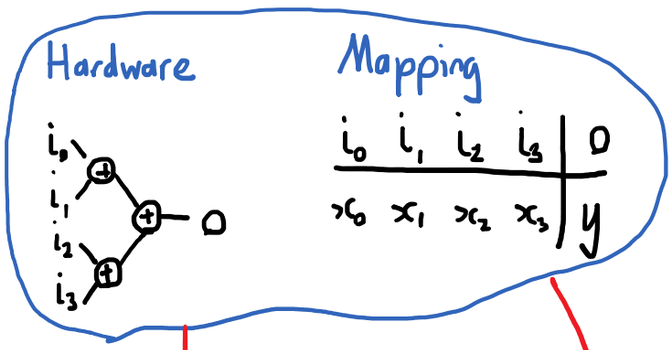
For any function there is a simple corresponding combinatorial circuit.
The mapping from the function inputs and outputs to the circuit inputs and outputs is direct.

We can make transformations of the (hardware, mapping) pair into a form that doesn't map as directly on the function.

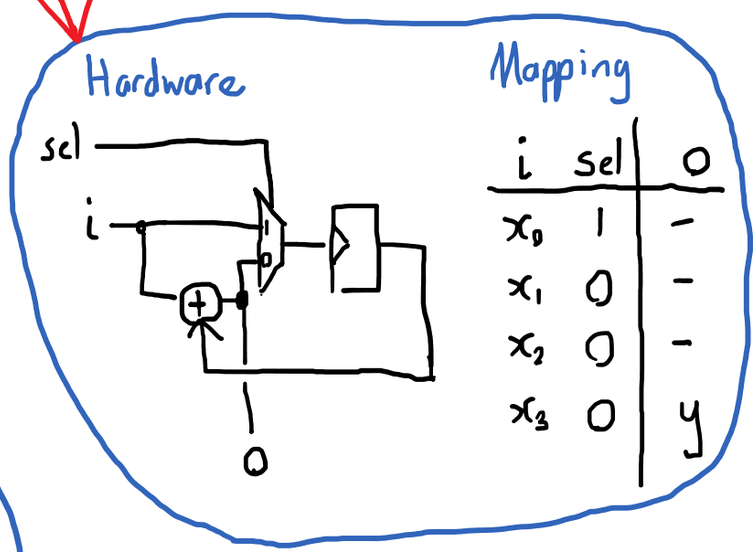
In this example we're changing from feeding in all 4 inputs on the same clock cycle to feeding them in over 2 clock cycles.

Adding 4 Numbers (2)

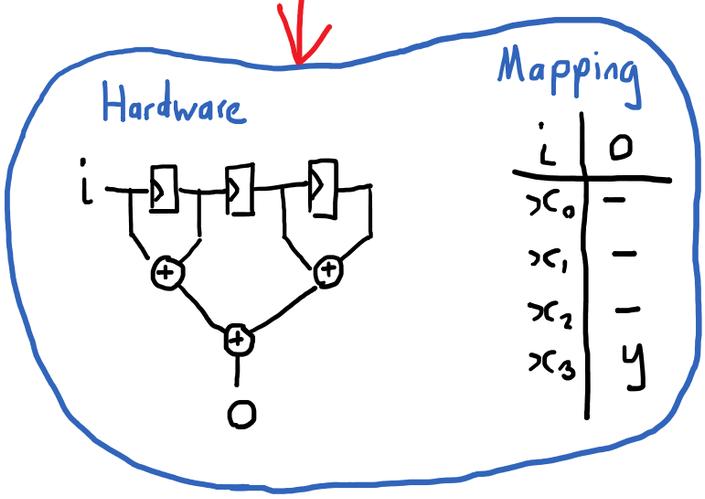
Monday, May 9, 2022 3:15 PM



More complex transformations can introduce control signals into the mapping.



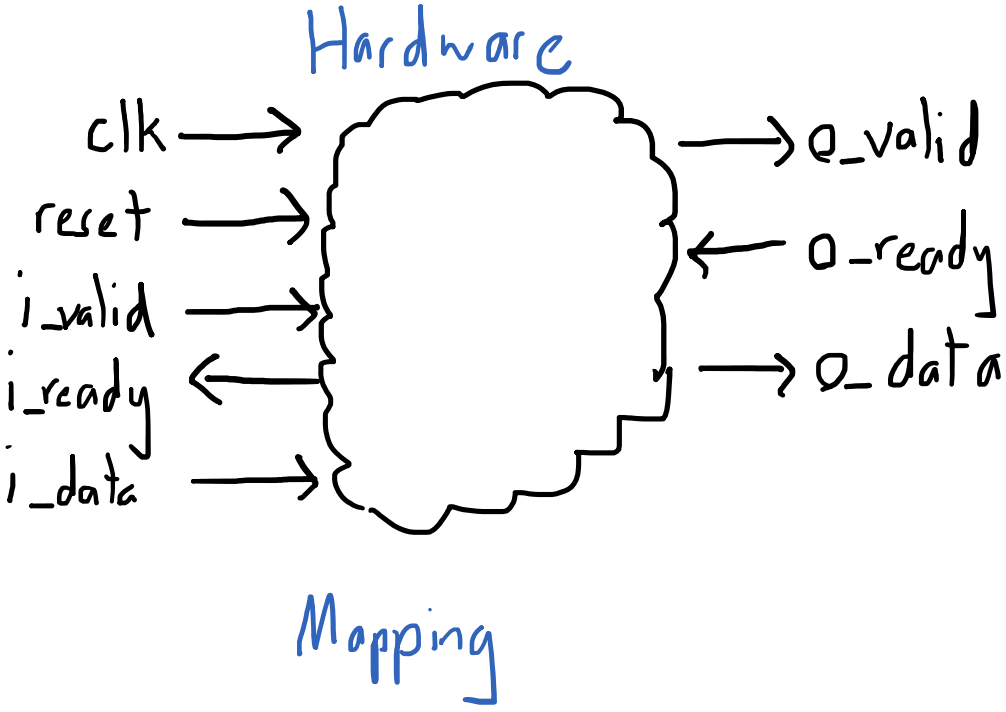
A simple inefficient mapping to use 1 input per clock cycle.



Adding 4 Numbers (3)

Monday, May 9, 2022 3:33 PM

Once we start getting more complex interface such as valid/ready handshaking the mapping tables that we've been showing become insufficient.
Now the mapping becomes a program itself where which algorithm inputs and outputs correspond to which hardware ports can depend on the input and output control signals of the hardware.



x_0 is the content of i_data after reset the first time i_valid and i_ready are both high.
x_1 is the content of i_data after reset the second time i_valid and i_ready are both high.
…
y is the content of o_data after reset the first time o_valid and o_ready are both high.

i_valid is unconstrained
o_ready is unconstrained
reset is unconstrained

Hardware reuse visualization

Monday, May 9, 2022 3:46 PM