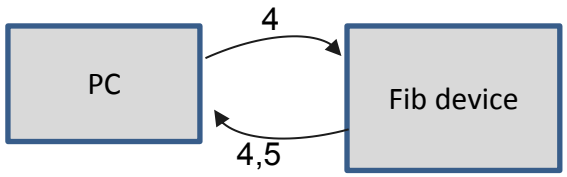
Lecture 11 - Short Exercise

Think of designing an application accelerator for calculating Fibonacci numbers.

$$\text{fib}(n) = \begin{cases} 0 & \text{if } n = 0 \\ 1 & \text{if } n = 1 \\ \text{fib}(n-1) + \text{fib}(n-2) & \text{if } n >= 2 \end{cases}$$

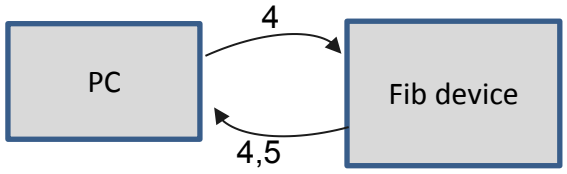


What sort of design pattern would suite such a device? Considering there would be a PC sending requests 1000s of requests and receiving paired results (input:output) in an unblocking manner (as illustrated above).

If you went the way of a designing a processor core to do this and have multiple of these cores on the digital accelerator, what instructions would each core execute? What other parts would be needed to make it a functional system?
Do some rough diagrams and discuss with your class mates.

Think of designing an application accelerator for calculating Fibonacci numbers.

$$\text{fib}(n) = \begin{cases} 0 & \text{if } n = 0 \\ 1 & \text{if } n = 1 \\ \text{fib}(n-1) + \text{fib}(n-2) & \text{if } n >= 2 \end{cases}$$

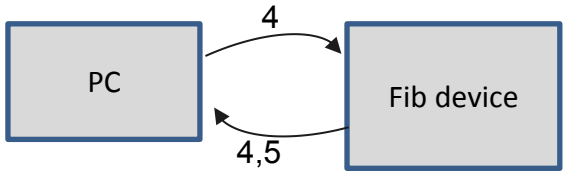


What sort of design pattern would suite such a device? Considering there would be a PC sending requests 1000s of requests and receiving paired results (input:output) in an unblocking manner (as illustrated above).

If you went the way of a designing a processor core to do this and have multiple of these cores on the digital accelerator, what instructions would each core execute? What other parts would be needed to make it a functional system?
Do some rough diagrams and discuss with your class mates.

Think of designing an application accelerator for calculating Fibonacci numbers.

$$\text{fib}(n) = \begin{cases} 0 & \text{if } n = 0 \\ 1 & \text{if } n = 1 \\ \text{fib}(n-1) + \text{fib}(n-2) & \text{if } n >= 2 \end{cases}$$



What sort of design pattern would suite such a device? Considering there would be a PC sending requests 1000s of requests and receiving paired results (input:output) in an unblocking manner (as illustrated above).

If you went the way of a designing a processor core to do this and have multiple of these cores on the digital accelerator, what instructions would each core execute? What other parts would be needed to make it a functional system?
Do some rough diagrams and discuss with your class mates.