Ready/valid interface

- Overview
  - The ready/valid interface is a useful way of letting one module pass resources to another module without having to know about its pipelining or throughput
  - Implemented by adding two additional connections alongside data – ‘DataReady’ and ‘DataValid’

- Turning the signals on
  - The producer of the resource sets its ‘DataValid’ signal high when the data on its datapath output is valid
  - The consumer of the resource sets its ‘DataReady’ signal high when it would be ready to copy in new data synchronously on the following positive of the clock
  - Values on cycle after reset depend on specification of the two modules!

- Moving data and turning the signals off
  - Data only moves on positive clock edges
  - Data is copied from the output stage of the producer to the input stage of the consumer on the first positive clock edge where both signals are high
  - On the clock cycle after the first cycle where both are high, ‘DataValid’ is only high if the producer has put new valid data on the datapath output
  - On the clock cycle after the first cycle where both are high, ‘DataReady’ is only high if the consumer is ready to receive another input

Problem 1:

Assume your list processor module from Lab 3 implements the ready/valid handshake (as this interface is known). It outputs the total sum of a list only once it reaches the end of the list and processes the sum. Fill in the following waveforms for the interface signals assuming exactly one sum is computed.