

Exercise 5.1.

Design a register machine to compute factorial using the iterative algorithm specified by the following procedure. Draw data-path and controller diagrams for this machine.

```
(define (factorial n)
  (define (iter product counter)
    (if (> counter n)
        product
        (iter (* counter product)
              (+ counter 1))))
  (iter 1 1))
```

Answer.

A machine to carry out this factorial algorithm must keep track of two numbers, *product* and *counter*, and we assume that these numbers are stored in two registers with those names. The basic operation required are testing whether contents of register *counter* exceeds the constant *n*, computing the product of the contents of register *counter* multiplied by the contents of register *product*, and computing the increment of the contents of register *counter*. On each cycle of the factorial algorithm, the contents of register *product* must be replaced by the product produced by multiplied itself to the contents of register *counter*, and the contents of register *counter* must be replaced by its increment. Figure 1 shows the data-path diagram for this machine, and figure 2 describes its controller respectively.

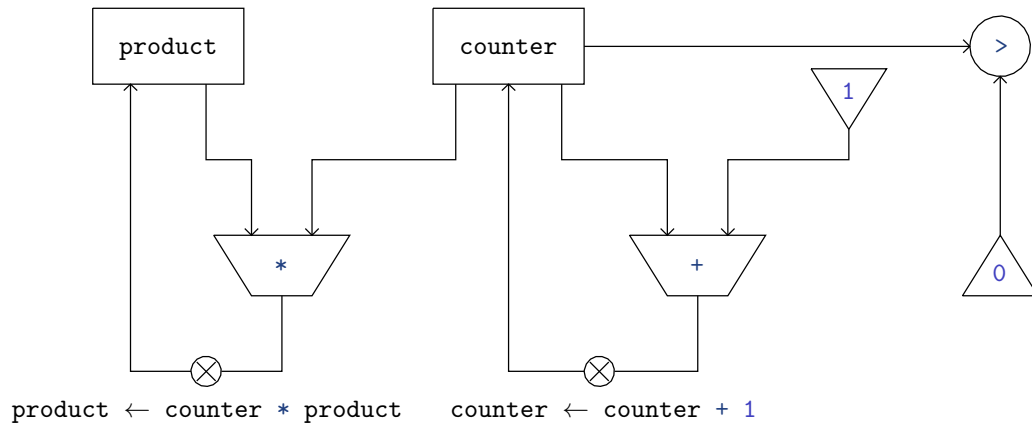



Figure 1. Data paths for a factorial machine.

*. Creative Commons  2014, Lawrence X. A. Yan (颜世敏, aka 颜序).
Email address: informlarry@gmail.com

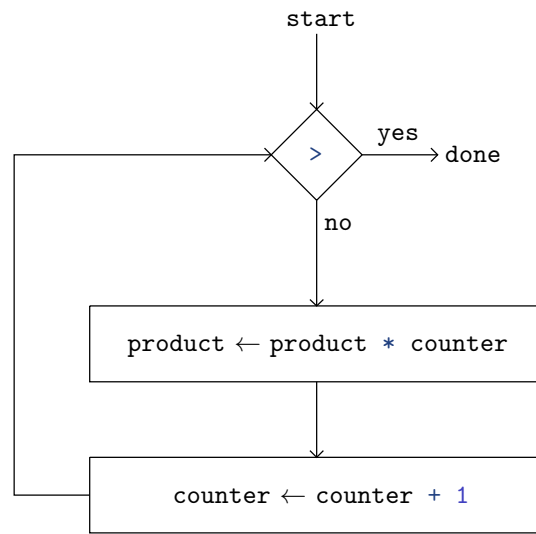


Figure 2. Controller for a factorial machine.