

# 1 Simulator für digitale Schaltkreise

Schreiben Sie einen Simulator, der das zeitliche Verhalten digitaler Schaltkreise simuliert. Die zu Simulierende Schaltung koennen Sie in Ihrer main Methode "aufbauen". Sie werden für Ihre Realisierung folgende Klassen brauchen:

## 1.1 EventCalender

- `void performAfter(TimeType delta, Event e)`
- `void propagate()`

## 1.2 Wire

Wires sollen nur den Zustände 0 und 1 übertragen. Ein Wire überträgt den Zustand, der von einer Component mittels eines Ausgangs aufgeprägt wird ohne Verzögerung an die Components, die mit einem Eingang mit dem Wire verbunden sind.

- `addInput(Component c)`
- `setState(int s)`
- `int getState()`

## 1.3 Component

Objekte vom Typ Component (abstrakte Klasse) sollen in den EventCalender eingetragen werden können.

- `InputAction()` wird von einem Wire aufgerufen, um der Component mitzuteilen, dass sich einer ihrer Eingänge verändert hat.
- Was immer nötig ist, damit Components in den EventCalender eingetragen werden können.

## 1.4 Konkrete Components

### 1.4.1 NandGate

Die Schaltverzögerung muss modelliert werden.

### 1.4.2 Clock

Ein Taktgenerator.

### 1.4.3 DFlipFlop

Setup- und Holdzeit müssen überwacht werden.

### 1.4.4 Force

Eine Force Component bekommt eine Liste von Paaren bestehend aus Zeitpunkten und Logikwerten, und gibt dieses Muster auf ihrem Ausgang aus. Damit kann eine Schaltung mit Testmustern stimuliert werden.

#### **1.4.5 Probe**

Eine Probe hat einen Eingang. Ändert dieser seinen Wert druckt die Probe ihren Namen, die Simulationszeit, und den neuen Wert des Eingangs aus. Damit kann eine Schaltung beobachtet werden.