Princeton University
COS 217: Introduction to Programming Systems
Algorithms for Circuit Simulator (Compiled Version)

compileCircuit

```
Generate "save %sp, -96, %sp" into code array
For each flip flop index i...
   Call compileExpr
      Actual parameter: i's Expr
      Return value: reg
   Generate "st reg, [%i2 + 4*i]" into code array
   Release register to register pool
Generate "ret" in code array
Generate "restore" in code array
```

compileExpr

```
Formal parameter: an Expr
Return value: reg

NUMBER Expr
   Fetch the number from the Expr
   Get a reg from the register pool
   Generate "mov number, reg" into code array
   Return reg

INPUTREF Expr
   Fetch input index from Expr
   Get a reg from the register pool
   Generate "ld [%i0 + index*4], reg" into code array
   Return reg

FLIPFLOPREF Expr
   Fetch flip flop index from Expr
   Get a reg from the register pool
   Generate "ld [%i1 + index*4], reg" into code array
   Return reg

NOT Expr
   Recursively compile Expr; note the reg
   Generate "xor reg, 1, reg" into code array
   Return reg

AND Expr
   Recursively compile left Expr; note the reg1
   Recursively compile right Expr; note the reg2
   Generate "and reg1, reg2, reg1" into code array
   Release reg2 to the register pool
   Return reg1

OR Expr
   Recursively compile left Expr; note the reg1
   Recursively compile right Expr; note the reg2
   Generate "or reg1, reg2, reg1" into code array
   Release reg2 to the register pool
   Return reg1
```