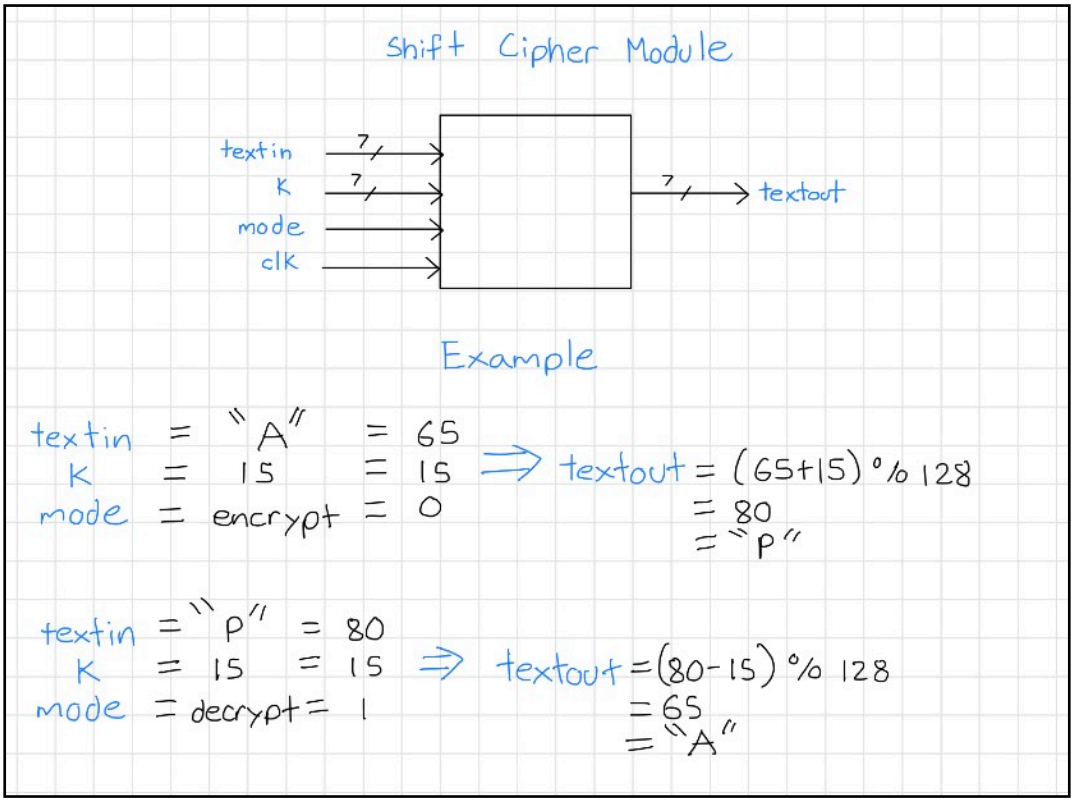
Shift Cipher Using Chisel

Design the sequential circuit shown below using Chisel. The circuit is a shift cipher that encrypts/decrypts one letter at a time. It receives a 7-bit input “textin” and produces a 7-bit output “textout”. These signals represent one of the 128 ASCII characters. This means the key is a 7-bit input. The circuit encrypts if “mode” is zero and decrypts if “mode” is one. Create a test function to test that the circuit works.




```
test(new ShiftCipher) { a =>
    // test encryption
    a.io.textin.poke(65.U)      // "A"
    a.io.key.poke(15.U)
    a.io.mode.poke(0.U)         // encrypt
    a.clock.step(1)
    a.io.textout.expect(80.U)   // "P"

    // test decryption
    a.io.textin.poke(80.U)      // "P"
    a.io.key.poke(15.U)
    a.io.mode.poke(1.U)         // decrypt
    a.clock.step(1)
    a.io.textout.expect(65.U)   // "A"
}
println("SUCCESS!!")
```

```
Elaborating design...
Done elaborating.
test ShiftCipher Success: 0 tests passed in 4 cycles in 0.005329 seconds 750.61 Hz
SUCCESS!!
defined class ShiftCipher
```

```
test(new ShiftCipher) { b =>
    for (i <- 0 until 127) {
        for (j <- 0 until 127) {
            // test encryption
            b.io.textin.poke(i.U)
            b.io.key.poke(j.U)
            b.io.mode.poke(0.U)
            b.clock.step(1)
            b.io.textout.expect(((i+j)%128).U)

            // test decryption
            b.io.textin.poke(i.U)
            b.io.key.poke(j.U)
            b.io.mode.poke(1.U)
            b.clock.step(1)
            b.io.textout.expect(  ((((i-j)%128)+128)%128).U)
        }
    }
}
println("SUCCESS AGAIN!!")
```

```
Elaborating design...
Done elaborating.
test ShiftCipher Success: 0 tests passed in 4 cycles in 0.004784 seconds 836.11 Hz
SUCCESS!!
Elaborating design...
Done elaborating.
test ShiftCipher Success: 0 tests passed in 32260 cycles in 3.210534 seconds 10048.17 Hz
SUCCESS AGAIN!!
defined class ShiftCipher
```