Applied Cryptography
CPEG 472/672
Lecture 2B

Instructor: Nektarios Tsoutsos

Real-World PRNGs

- PRNGs can be based on software libraries or hardware modules
- PRNG “devices” in Linux
  - /dev/urandom (non-blocking pool)
  - /dev/random (blocking pool)
- What is the difference?
  - /dev/random estimates the entropy left in the pool
  - However, entropy estimators are unreliable

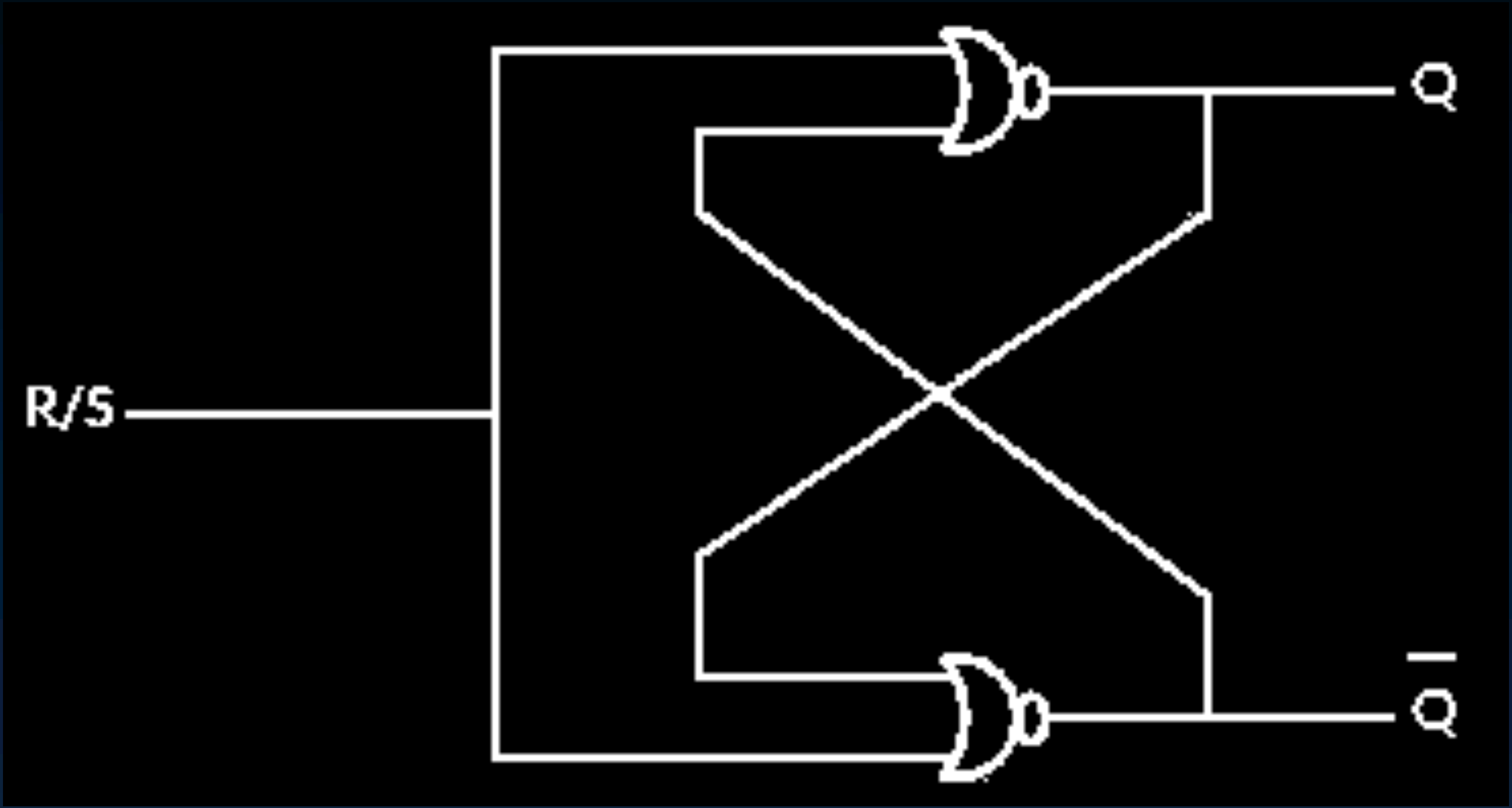
Real-World PRNGs

- PRNG function in Windows
  - CryptGenRandom (older)
  - BcryptGenRandom (new)
- Things can go wrong
  - Requires the user to acquire a cryptographic context, which could fail
  - Need special checks
  - Bug in TrueCrypt

Real-World PRNGs

- Intel’s Hardware PRNG
  - 2012 Ivy Bridge microarchitecture
  - Based on AES CTR DRBG (NIST SP 800-90)
  - RDRAND, RDSEED instructions
  - Single entropy source based on a metastable circuit
  - Affected by thermal noise fluctuations
  - Carry flag is set to 1 if the generated random data are valid

RS-NOR latch metastable circuit



Examples of PRNG failures

- Netscape (47bits entropy, not 128)

20 bits

```
RNG_CreateContext()
(seconds, microseconds) = time of day;
pid = process ID; ppid = parent process ID;
a = transform(microseconds);
b = transform(pid + seconds + (ppid << 12));
seed = MD5(a, b);
```

overlap

Examples of PRNG failures

- RSA primes generated at boot time

```
prng.seed(seed)
p = prng.generate_random_prime()
q = prng.generate_random_prime()
n = p*q
```

- Generates identical primes

Examples of PRNG failures

- RSA primes generated at boot time

```
prng.seed(seed)
p = prng.generate_random_prime()
prng.add_entropy()
q = prng.generate_random_prime()
n = p*q
```

- Generates the same p (how to recover?)

Examples of PRNG failures

- Cryptocat (uniform decimal digits)

```
Cryptocat.random = function() {
   var x, o = "";
   while (o.length < 16) {
      x = state.getBytes(1);
      if (x[0] <= 250) {
         o += x[0] % 10;
      }
   }
   return parseFloat('0.' + o)
}
```

What is the problem?

Is it fixed?

Reading for next lecture

- Aumasson: Chapter 4