Random Number Generator

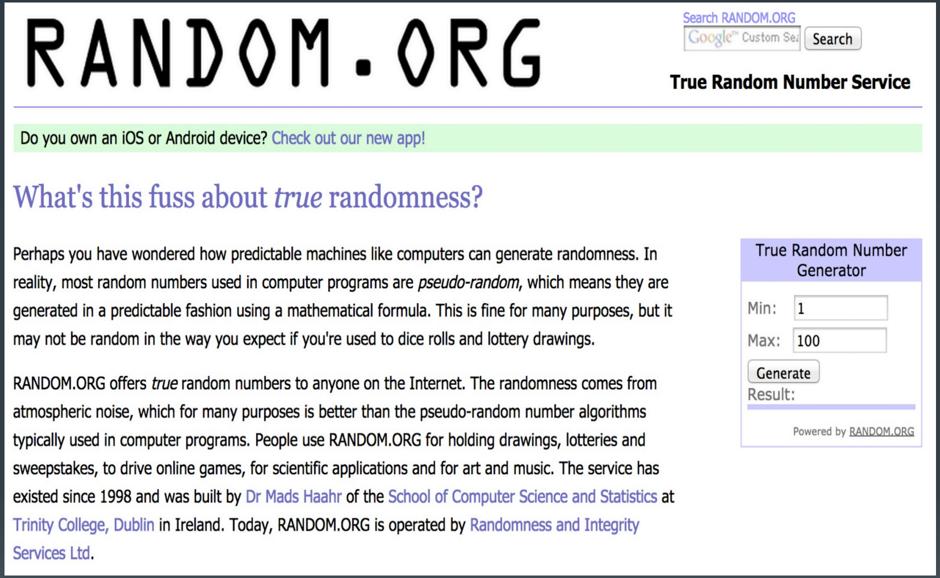
Long He

Design

Method I have tried:

www.random.org
Random.org is a website that produces “true” random numbers based on atmospheric noise.

Collect the noise in operating system: mouse movement and click, keyboard input, microphone.



Do you own an iOS or Android device? Check out our new app!

What's this fuss about *true* randomness?

Perhaps you have wondered how predictable machines like computers can generate randomness. In reality, most random numbers used in computer programs are *pseudo-random*, which means they are generated in a predictable fashion using a mathematical formula. This is fine for many purposes, but it may not be random in the way you expect if you're used to dice rolls and lottery drawings.

RANDOM.ORG offers *true* random numbers to anyone on the Internet. The randomness comes from atmospheric noise, which for many purposes is better than the pseudo-random number algorithms typically used in computer programs. People use RANDOM.ORG for holding drawings, lotteries and sweepstakes, to drive online games, for scientific applications and for art and music. The service has existed since 1998 and was built by Dr Mads Haahr of the School of Computer Science and Statistics at Trinity College, Dublin in Ireland. Today, RANDOM.ORG is operated by Randomness and Integrity Services Ltd.

True Random Number Generator

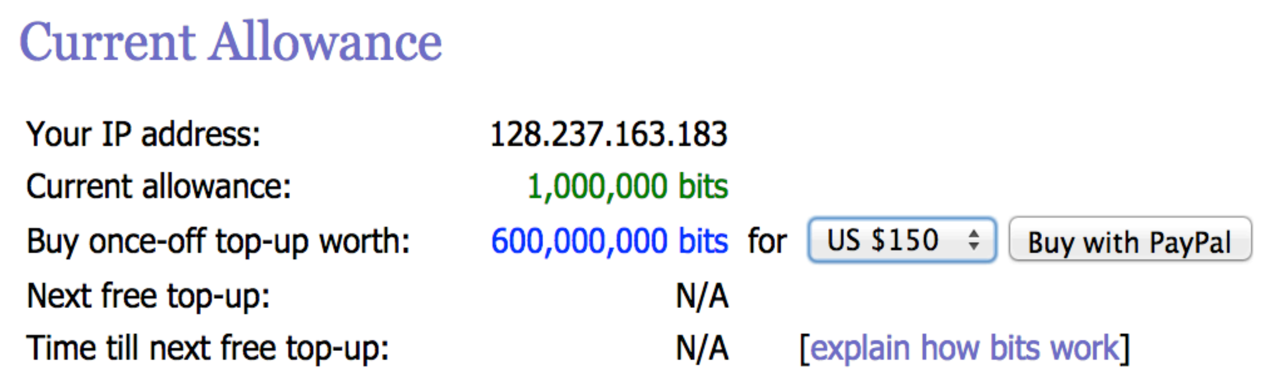
Min: 1

Max: 100

Why don’t I choose these method?

www.random.org

I have to pay for true random number.



Current Allowance

Your IP address:	128.237.163.183
Current allowance:	1,000,000 bits
Buy once-off top-up worth:	600,000,000 bits for US $150 Buy with PayPal
Next free top-up:	N/A
Time till next free top-up:	N/A [explain how bits work]

Noise in operating system

Low speed and easy to be exhausted.

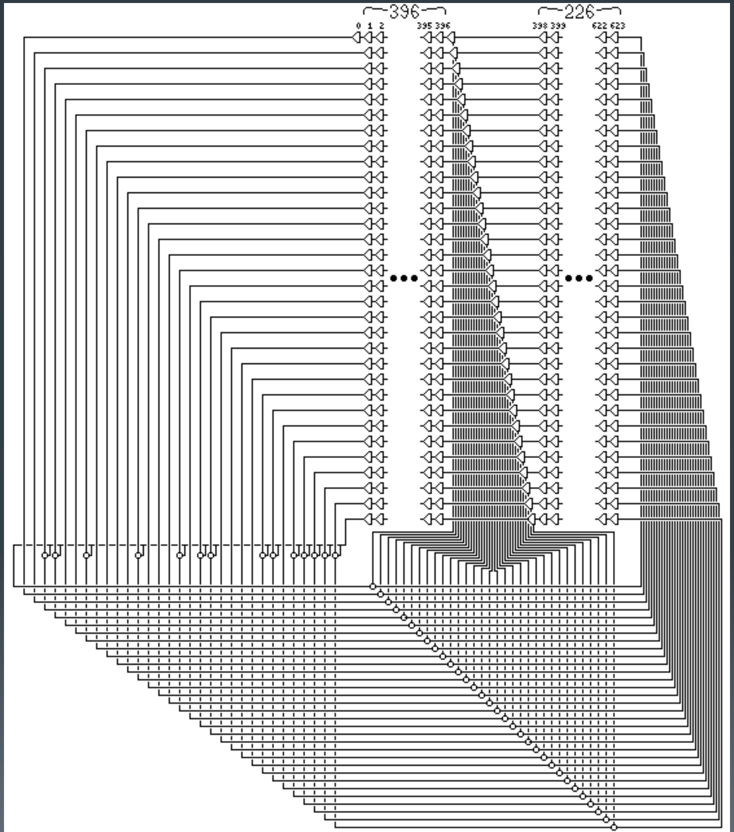
Comparative low precision of CPU timer based on Java.

Mersenne Twister

The state succession algorithm of the Mersenne Twister can be depicted in an illustration as a kind of linear-feedback shift register.

http://www.quadibloc.com/crypto/co4814.htm

http://www.cs.gmu.edu/~sean/research/



Design

Why I choose Mersenne Twister:

One of the best pseudorandom number generator

High speed (more details later)

Low bias

Long period period: 2^19937–1(Mersenne Prime)

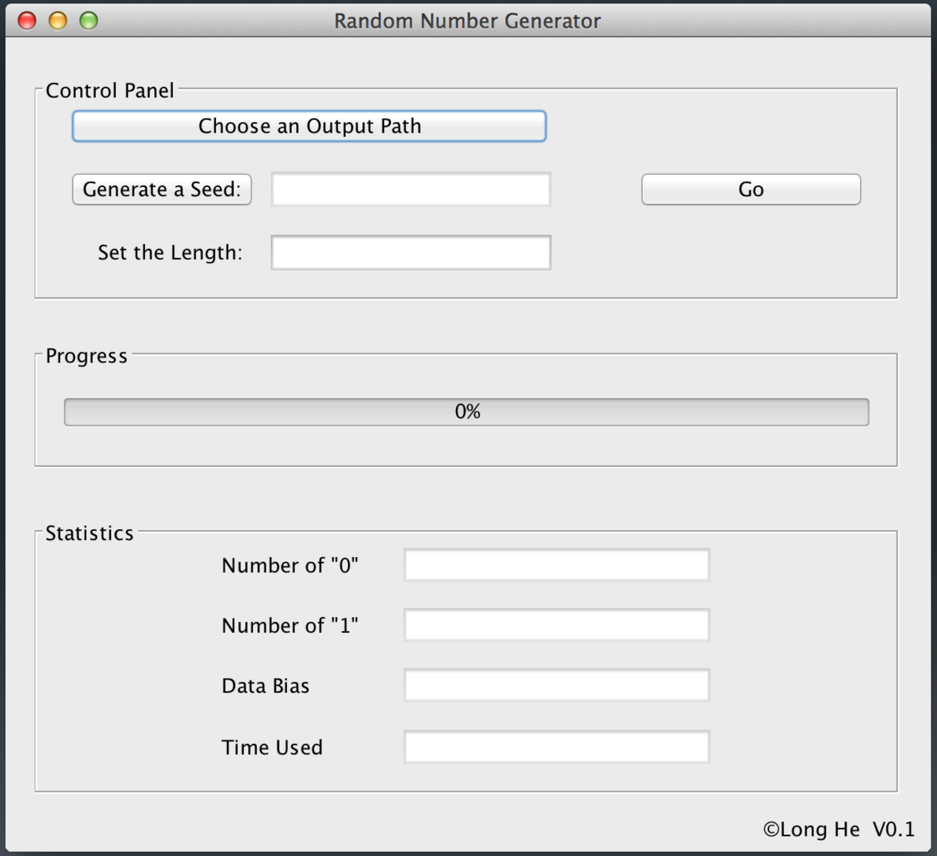
Design Implement

Multithreading

Progress Bar

Java.math.BigInteger

System.nanoTime()



Random Number Generator

Control Panel

Choose an Output Path

Generate a Seed:

Go

Set the Length:

Progress

0%

Statistics

Number of "0"

Number of "1"

Data Bias

Time Used

©Long He V0.1

Multithreading

Show the number of “0”, number of “1”, data bias in the process dynamically.

Progress Bar

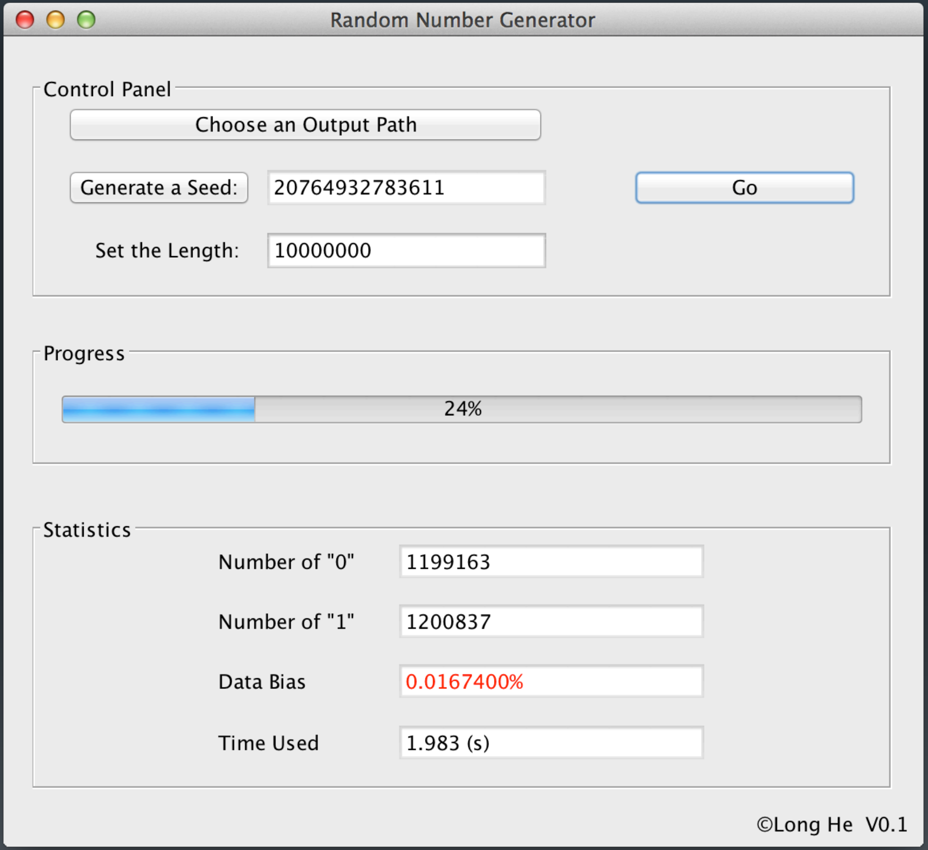
Show the progress rate.

Java.math.BigInteger

For extreme long input.

System.nanoTime()

Extreme high precision for seed generating.



Generate a Seed: 20764932783611

Set the Length: 10000000

Number of "0" 1199163

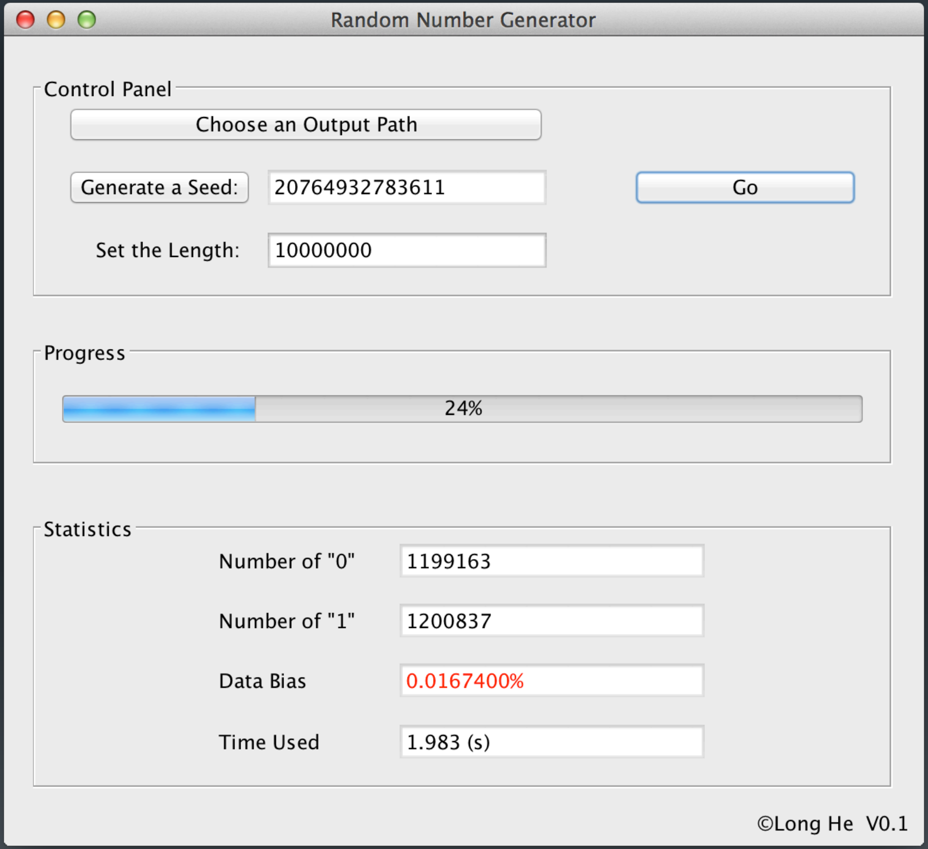
Number of "1" 1200837

Data Bias 0.0167400%

Time Used 1.983 (s)

©Long He V0.1

Design Implement Usage



Random Number Generator

Control Panel

Generate a Seed: 20764932783611

Set the Length: 10000000

Progress

24%

Statistics

Number of "0" 1199163

Number of "1" 1200837

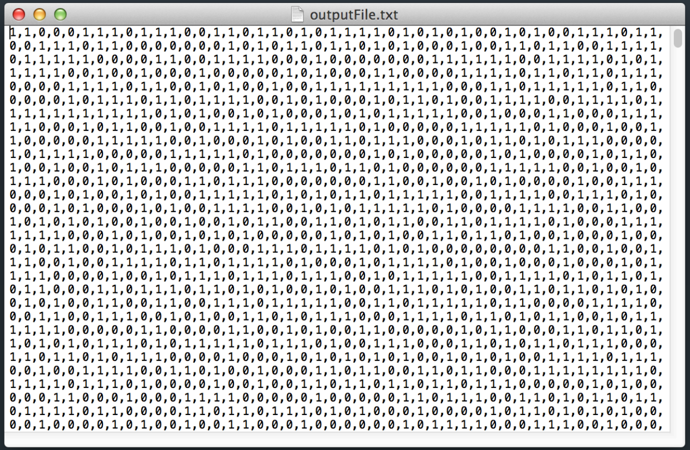
Data Bias 0.0167400%

Time Used 1.983 (s)

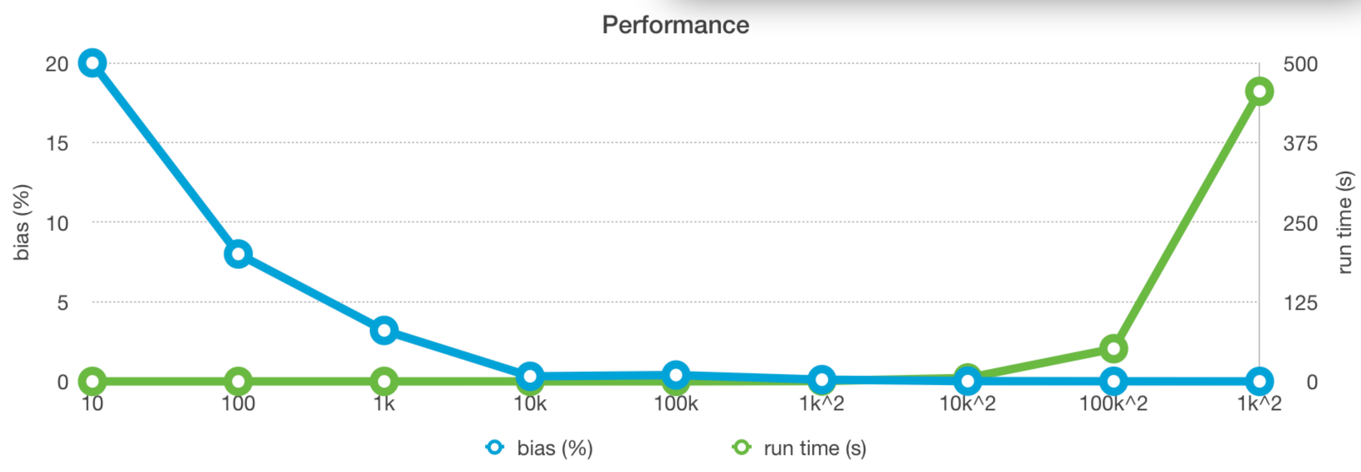
©Long He V0.1

Design Implement Usage Performance

Trial	bias (%)	run time (s)
10	20	0.001
100	8	0.002
1k	3.2	0.002
10k	0.32	0.009
100k	0.4	0.056
1k^2	0.11	0.585
10k^2	0.018	5.459
100k^2	0.0085	51.366
1k^3	0.0022	455.416



```
1,1,0,0,0,1,1,1,0,1,1,1,0,0,1,1,0,1,1,0,1,0,1,1,1,1,0,1,0,1,0,1,0,0,1,0,1,0,0,1,1,1,0,1,1,
0,0,1,1,1,0,1,1,0,0,0,0,0,0,0,1,0,1,0,1,1,0,1,1,0,1,0,1,0,0,0,1,0,0,1,1,0,1,1,0,0,1,1,1,1,
0,1,1,1,1,1,0,0,0,0,1,1,0,0,1,1,1,1,0,0,0,1,0,0,0,0,0,0,0,1,1,1,1,1,1,0,0,1,1,1,1,0,1,0,1,
1,1,1,1,0,0,1,0,0,1,0,0,0,1,0,0,0,0,0,1,0,1,0,0,0,1,1,0,0,0,0,1,1,1,1,0,1,1,0,1,1,0,1,1,1,
0,0,0,0,1,1,1,1,0,1,1,0,0,1,0,1,0,0,1,0,0,1,1,1,1,1,1,1,1,1,0,0,0,1,1,0,1,1,1,1,1,0,1,1,0,
0,0,0,0,1,0,1,1,1,0,1,1,0,1,1,1,1,0,0,1,0,1,0,0,0,1,0,1,1,0,1,0,0,1,1,1,1,0,0,1,1,1,1,0,1,
1,1,1,1,1,1,1,1,1,1,0,1,0,1,0,0,1,0,1,0,0,0,1,0,1,0,1,1,1,1,1,0,0,1,0,0,0,1,1,0,0,0,1,1,1,
1,1,0,0,0,1,0,1,1,0,0,1,0,0,1,1,1,1,0,1,1,1,1,1,0,1,0,0,0,0,0,1,1,1,1,1,0,1,0,0,0,1,0,0,1,
1,0,0,0,0,0,1,1,1,1,1,0,0,1,0,0,0,1,0,1,0,0,1,1,0,1,1,1,0,0,0,1,0,1,1,0,1,0,1,1,1,0,0,0,0,
1,0,1,1,1,1,0,0,0,0,0,1,1,1,1,1,0,1,0,0,0,0,0,0,0,1,0,1,0,0,0,0,0,1,0,1,0,0,0,0,1,0,1,1,0,
1,0,0,1,0,0,1,0,1,1,1,0,0,0,0,0,1,1,0,1,1,1,0,1,1,0,1,0,0,0,0,0,0,1,1,1,1,1,0,0,1,0,0,1,0,
1,1,1,0,0,0,1,0,1,0,0,0,1,1,0,1,1,1,0,0,0,0,0,0,0,1,1,0,0,1,0,0,1,0,1,0,0,0,0,1,0,0,1,1,1,
0,0,0,1,0,1,0,0,1,0,1,0,0,1,1,1,1,1,0,1,0,1,0,1,1,0,1,1,1,1,1,0,0,1,1,1,1,0,0,1,1,1,0,1,0,
0,0,0,1,0,1,0,0,0,1,0,1,0,0,1,1,1,1,0,0,1,0,1,0,1,1,1,1,1,0,1,0,0,0,0,1,1,1,1,0,0,1,1,0,0,
1,0,1,0,1,0,1,0,0,1,0,0,1,0,0,1,0,1,1,0,0,1,1,0,1,0,1,1,0,0,1,1,0,1,1,1,1,0,1,0,0,0,1,1,1,
1,1,1,1,0,0,0,1,0,1,0,0,1,0,1,0,1,0,0,0,0,0,1,0,1,0,1,0,0,1,1,0,1,1,0,1,0,0,1,0,0,0,1,0,0,
0,1,0,1,1,0,0,1,0,1,1,1,0,1,0,0,0,1,1,1,0,1,1,1,1,0,1,0,1,0,0,0,0,0,0,0,0,1,1,0,0,1,0,0,1,
1,1,0,0,1,0,0,1,1,1,1,0,1,1,0,1,1,1,1,0,1,0,0,0,1,0,1,1,1,1,0,1,0,0,1,0,0,0,1,0,0,0,1,0,1,
1,1,1,0,0,0,0,1,0,0,1,0,1,1,1,0,1,1,1,0,1,1,1,0,0,1,0,1,1,1,1,1,0,0,1,1,1,1,0,1,0,1,1,0,1,
0,1,1,0,0,0,1,1,0,1,1,1,0,1,1,0,1,0,1,0,0,1,0,1,0,0,1,1,1,0,1,0,1,0,0,1,1,0,1,1,0,1,0,1,0,
0,1,0,1,0,0,1,1,0,0,1,1,0,0,1,1,1,0,1,1,1,1,1,0,0,1,1,0,1,1,1,1,1,0,1,1,0,0,0,0,1,1,1,1,0,
0,0,1,1,0,0,1,1,1,0,0,1,0,1,0,0,1,1,0,1,0,1,1,1,0,0,0,1,1,1,1,0,1,1,0,1,0,1,1,0,0,1,0,1,1,
1,1,1,1,0,0,0,0,0,1,1,0,0,0,0,1,1,0,0,1,0,1,0,0,1,1,0,0,0,0,0,1,0,1,1,0,0,0,1,1,0,1,1,0,1,
1,0,1,0,1,0,1,1,1,0,1,0,1,1,1,1,1,0,1,1,1,0,1,0,0,1,1,1,0,0,0,1,1,0,1,0,1,1,0,1,1,1,0,0,0,
1,1,0,1,1,0,1,0,1,1,1,0,0,0,0,1,0,0,0,1,0,1,0,1,0,1,0,1,0,0,1,0,1,0,1,0,0,1,1,1,1,0,1,1,1,
0,0,1,0,0,1,1,1,1,0,0,1,1,0,1,0,0,1,0,0,0,1,1,0,1,1,0,0,1,1,0,1,1,0,0,0,1,1,1,1,1,1,1,1,0,
1,1,1,1,0,1,1,1,0,1,0,0,0,0,1,0,0,1,0,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,1,0,0,0,0,0,1,0,1,0,0,
0,0,0,1,1,0,0,0,1,0,0,0,1,1,1,1,0,0,0,0,0,1,0,0,0,0,0,1,1,0,1,1,1,0,0,1,1,0,1,0,1,1,0,1,1,
0,1,1,1,1,0,1,1,0,0,0,0,1,1,0,1,1,0,1,1,1,0,1,0,1,0,0,0,1,0,0,0,0,1,0,1,1,0,1,0,1,0,1,0,0,
0,0,1,0,0,0,0,1,0,1,0,0,1,0,0,1,1,0,0,0,1,0,0,0,0,0,0,1,0,1,1,1,1,0,0,0,1,1,1,0,0,1,0,0,0,
```



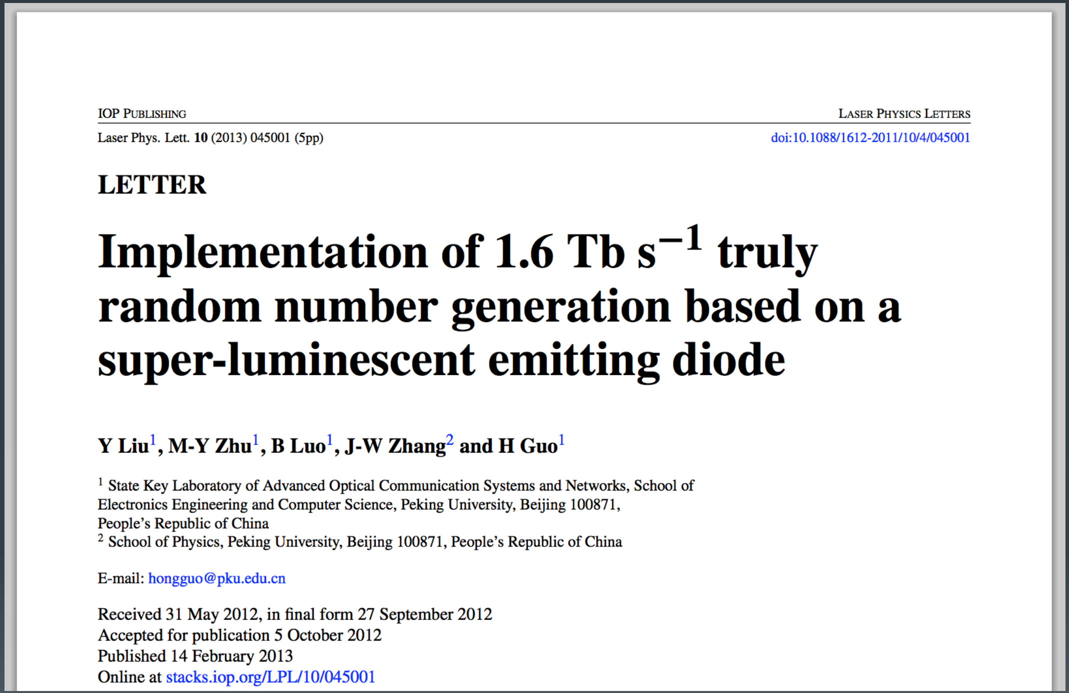
My Future Idea:

Utilize laser or optical mouse to generate random binary numbers.

Advantages:

Real random number

High speed



IOP PUBLISHING LASER PHYSICS LETTERS

Laser Phys. Lett. **10** (2013) 045001 (5pp) doi:10.1088/1612-2011/10/4/045001

LETTER

Implementation of 1.6 Tb s^{-1} truly random number generation based on a super-luminescent emitting diode

Y Liu[1], M-Y Zhu[1], B Luo[1], J-W Zhang[2] and H Guo[1]

[1] State Key Laboratory of Advanced Optical Communication Systems and Networks, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, People's Republic of China

[2] School of Physics, Peking University, Beijing 100871, People's Republic of China

E-mail: hongguo@pku.edu.cn

Received 31 May 2012, in final form 27 September 2012
Accepted for publication 5 October 2012
Published 14 February 2013
Online at stacks.iop.org/LPL/10/045001

Paper on *Laser Physics Letters*:

http://iopscience.iop.org/1612-202X/10/4/045001/pdf/lpl13_4_045001.pdf

Thank you.