

Re: Hardware True Random Number Generator design / concept

Source: <http://sci.tech-archive.net/Archive/sci.electronics.design/2005-06/msg03092.html>

- *From:* Robert Baer <robertbaer@xxxxxxxxxxxxxx>
 - *Date:* Thu, 23 Jun 2005 05:52:10 GMT
-

The Real Andy wrote:

On Tue, 21 Jun 2005 16:42:07 -0700, Yoy G0 <yoyg0@xxxxxxxxxxxxxx> wrote:

-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA1

On Wed, 22 Jun 2005 06:49:45 +1000, you wrote:
[snip]

Have you considered using a Psuedo RNG with hardware entropy for seeding? There is plenty of great information out there to do this, and it saves on having to buy hardware. Do a google for Mersenne Twister, very good algorithm, long cycle.

By the way, a good statistical package for testing is R. Its free and its very powerful.

I was thinking about a possibility of using that method for producing key material for One Time Pad (OTP - cryptography), but Matt Mahoney ommented in sci.crypt so (POSTING - Re: Secure Data & Communication Project):

That is not one time pad. Not that it can't be done securely, but you don't have the theoretical secrecy against an attacker with unlimited computing power that OTP offers. With unlimited power the attacker can try all possible seeds (since there are only a finite number of them)

Re: Hardware True Random Number Generator design / concept

and find the one that decrypts to something sensible.
All the wrong decryptions will look like random data.
With OTP all plaintexts are equally likely,
including all the sensible ones, so there is no way to
tell which one is correct.
OTP will require a hardware random number
generator for every bit of the keystream.

Given any number of unlimited resources, one can crack any
cryptographic system. You need to determine your requirements and then
make a decision based on how much money you want to spend and how much
development time you wish to put in and how secure you require the
system to be.

Also, I fo