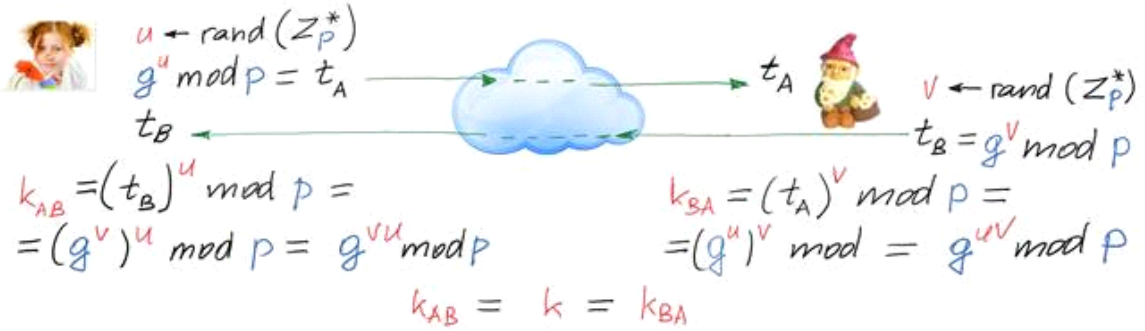
Diffie-Hellman Key Agreement Protocol (DH KAP)

Public Parameters PP=(p, g)



Discrete Logarithm Problem - DLP

$dlog_g\, t_A = dlog_g\,(g^u \bmod p) = u\, dlog_g\,(g \bmod p) = u \cdot 1 = u.$

Eo : after receiving t_A and computed u by solving DLP, compromizes common secret key k between A and B

Eo : computes $k = (t_A)^u$.

The **decisional Diffie–Hellman (DDH) assumption** is a computational hardness assumption about a certain problem involving discrete logarithms in cyclic groups. It is used as the basis to prove the security of many cryptographic protocols, most notably the ElGamal and Cramer–Shoup cryptosystems.

From <https://en.wikipedia.org/wiki/Decisional_Diffie%E2%80%93Hellman_assumption>

Definition [edit]

Consider a (multiplicative) cyclic group G of order q, and with generator g. The DDH assumption states that, given g^a and g^b for uniformly and independently chosen $a, b \in \mathbb{Z}_q$, the value g^{ab} "looks like" a random element in G.

This intuitive notion can be formally stated by saying that the following two probability distributions are computationally indistinguishable (in the security parameter, $n = \log(q)$): number of bits used in cryptosystem

- (g^a, g^b, g^{ab}), where a and b are randomly and independently chosen from $\mathbb{Z}_q$.
- (g^a, g^b, g^c), where a, b, c are randomly and independently chosen from $\mathbb{Z}_q$.

Triples of the first kind are often called **DDH triplet** or **DDH tuples**.

Indistinguishability means that it is infeasible to distinguish g^{ab} from $g^c \bmod p$.

Discrete Exponent Function (13/14) $DFF_{g,p}(x) = g^x \bmod p$

For illustration of 1-to-1 mapping of **DEF**$_7$(x) we perform the following step-by-step computations.

$p = 11$ & $g = 7$

	$x \in Z_{10}$	$a \in Z_{11}^*$
$7^0 = 1 \bmod 11$	0	1
$7^1 = 7 \bmod 11$	1	2
$7^2 = 5 \bmod 11$	2	3
$7^3 = 2 \bmod 11$	3	4
$7^4 = 3 \bmod 11$	4	5
$7^5 = 10 \bmod 11$	5	6
$7^6 = 4 \bmod 11$	6	7
$7^7 = 6 \bmod 11$	7	8
$7^8 = 9 \bmod 11$	8	9
$7^9 = 8 \bmod 11$	9	10

$DEF_g(x)$ 1-to-1 function
bijective — " —

It is seen that one value of x is mapped to one value of a.

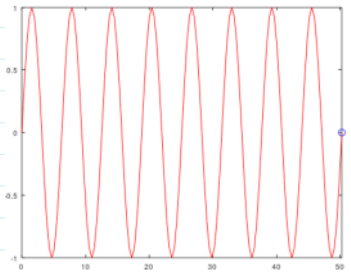
Discrete Exponent Function (14/14)

But the most in interesting think is that **DEF** is behaving like a *pseudorandom function*.

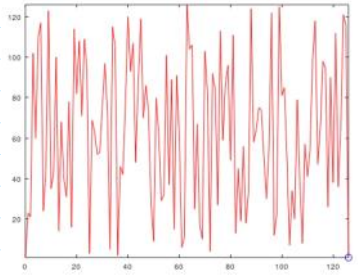
It is a main reason why this function is used in cryptography - classical cryptography.

To better understand the pseudorandom behaviour of **DEF** we compare the graph of "regular" **sine** function with "pseudorandom" **DEF** using Octave software.

```
>> p128sin
xrange = 16 * pi;
step = xrange/128;
x = 0:step:xrange;
y = sin(x);
comet(x, y)
```



```
>> p128def
p = 127;
g = 23;
x = 0:p-1;
a = mod_expv(g, x, p);
comet(x, a)
```



Groups for which DDH is assumed to hold [edit]

When using a cryptographic protocol whose security depends on the DDH assumption, it is important that the protocol is implemented using groups where DDH is believed to hold:

- The subgroup $\mathbb{G}_q$ of k-th residues modulo a prime $p = kq + 1$, where q is also a large prime (also called a Schnorr group). For the case of $k = 2$, this corresponds to the group of quadratic residues modulo a safe prime.
- The quotient group $\mathbb{Z}_p^*/\{1, -1\}$ for a safe prime $p = 2q + 1$, which consists of the cosets $\{\{1, -1\}, \ldots \{q, -q\}\}$. These cosets $\{x, -x\}$ can be represented by x, which implies $\mathbb{Z}_p^*/\{1, -1\} \equiv \{1, \ldots, q\}$. Since $\mathbb{Z}_p^*/\{1, -1\}$ and $\mathbb{G}_q$ are isomorphic, and the isomorphism can be computed efficiently in both direction, DDH is equally hard in both groups.
- A prime-order elliptic curve E over the field $GF(p)$, where p is prime, provided E has large embedding degree.
- A Jacobian of a hyper-elliptic curve over the field $GF(p)$ with a prime number of reduced divisors, where p is prime, provided the Jacobian has large embedding degree.

Importantly, the DDH assumption **does not hold** in the multiplicative group $\mathbb{Z}_p^*$, where p is prime. This is because if g is a generator of $\mathbb{Z}_p^*$, then the Legendre symbol of g^a reveals if a is even or odd. Given g^a, g^b and g^{ab}, one can thus efficiently compute and compare the least significant bit of a, b and ab, respectively, which provides a probabilistic method to distinguish g^{ab} from a random group element.

The DDH assumption does not hold on elliptic curves over $GF(p)$ with small embedding degree (say,

For example G_q when q=5, and p=2*q +1 = 11. G_{11} is a subgroup of Z^*_{11} = {1, 2, 3, …, 10}
This subgroup has a great importance in cryptography we denote by

$$G_5 = \{1, 3, 4, 5, 9\}.$$

The multiplication table of G_5 elements extracted from multiplication table of Z_{11}^* is presented below.

Multiplication tab. mod 11	*G*5				
*	**1**	**3**	**4**	**5**	**9**
1	1	3	4	5	9
3	3	9	1	4	5
4	4	1	5	9	3
5	5	4	9	3	1
9	9	5	3	1	4

Values of inverse elements in G_5

$1^{-1} = 1 \bmod 11$
$3^{-1} = 4 \bmod 11$
$4^{-1} = 3 \bmod 11$
$5^{-1} = 9 \bmod 11$
$9^{-1} = 5 \bmod 11$

Exponent tab. mod 11	*G*5					
^	**0**	**1**	**2**	**3**	**4**	**5**
1	1	1	1	1	1	1
3	1	3	9	5	4	1
4	1	4	5	9	3	1
5	1	5	3	4	9	1
9	1	9	4	3	5	1

In the case of Ecryption metod, either symmetric or asymmetric the indistiguishability assumption means that

$E(k, M) = C$ $\quad$ $Enc(a, m) = c$ // $m < p$

Indistinguishable from random bit string when C & c are transformed to bit string.

Indistinguishable from random bit string
when G & c are transformed to bit string.

FIPS - 140 - 2.

Chips cloning technique

Integrated circuits can be compromised using Undetectable hardware Trojans

From <https://thehackernews.com/2013/09/Undetectable-hardware-Trojans.html>

M 1:120

A team of researchers from the U.S. and Europe has developed a Hardware Trojan, which is an undetectable to many techniques, raising the question on need of proper hardware qualification.
They released a paper on stealthy Dopant-Level Hardware Trojans, showing how integrated circuits used in computers, military equipment and other critical systems can be maliciously compromised during the manufacturing process.
"*In this paper we propose an extremely stealthy approach for implementing hardware Trojans below the gate level, and we evaluate their impact on the security of the target device. Instead of adding additional circuitry to the target design, we insert our hardware Trojans by changing the dopant polarity of existing transistors.*" states the paper abstract.

Personal Identification Chip that is about the size of a grain of rice and implanted under the skin
NFC - Near Field Communication.

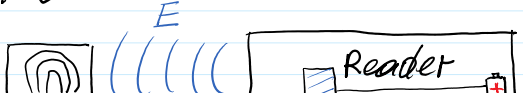
Would YOU let your boss implant you with a microchip? Belgian firm offers to turn staff into cyborgs to replace ID cards.

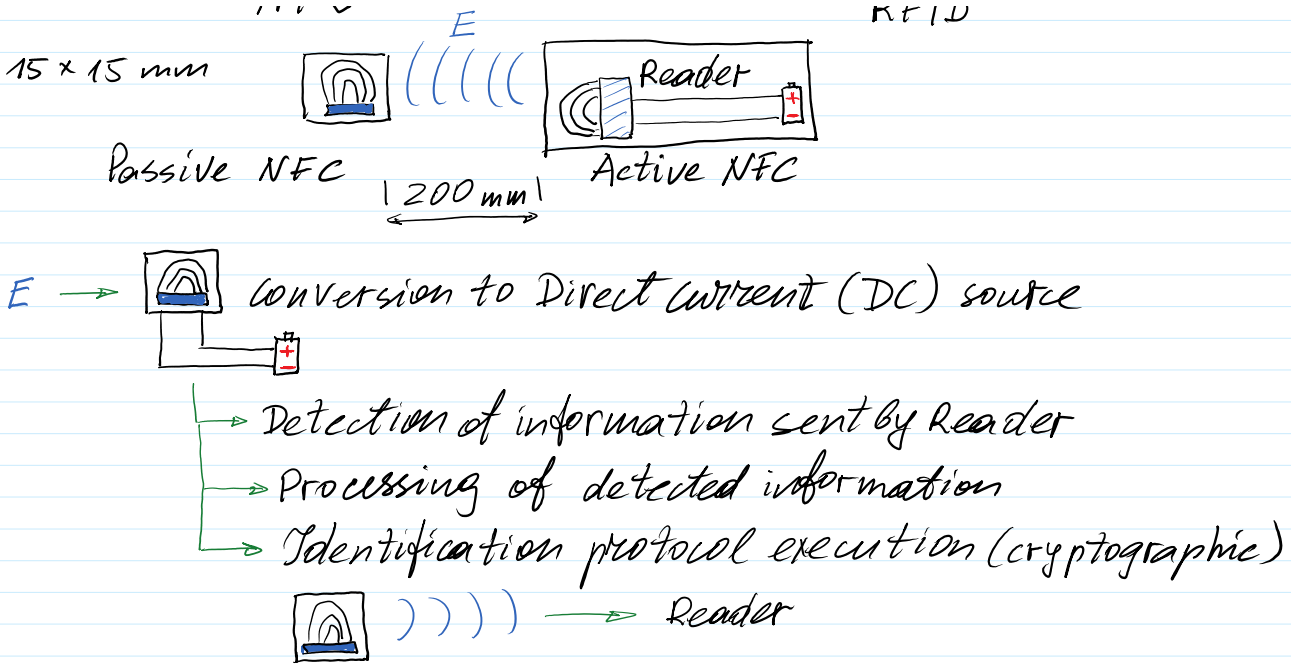
Read more: http://www.dailymail.co.uk/sciencetech/article-4203148/Company-offers-RFID-microchip-implants-replace-ID-cards.html#ixzz57Z0aeYgj
Follow us: @MailOnline on Twitter | DailyMail on Facebook

From <http://www.dailymail.co.uk/sciencetech/article-4203148/Company-offers-RFID-microchip-implants-replace-ID-cards.html>

Near Field communication NFC — Radio Frequency Identification RFID

15 x 15 mm





INTRINSIC ID

***Intrinsic* ID SRAM PUF Technology & Solutions: Physically Unclonable Function**

Intrinsic ID delivers strong, device-unique data security and authentication solutions for the connected world. These authentication solutions are based on Intrinsic ID's patented SRAM (Static Random Access Memory) Physical Unclonable Function or SRAM PUF technology.

Using this technology, security keys and unique identifiers can be extracted from the innate characteristics of each semiconductor. Similar to biometrics measures, these identifiers cannot be cloned, guessed, stolen or shared. Keys are generated only when required and don't remain stored on the system, hence providing the highest level of protection.

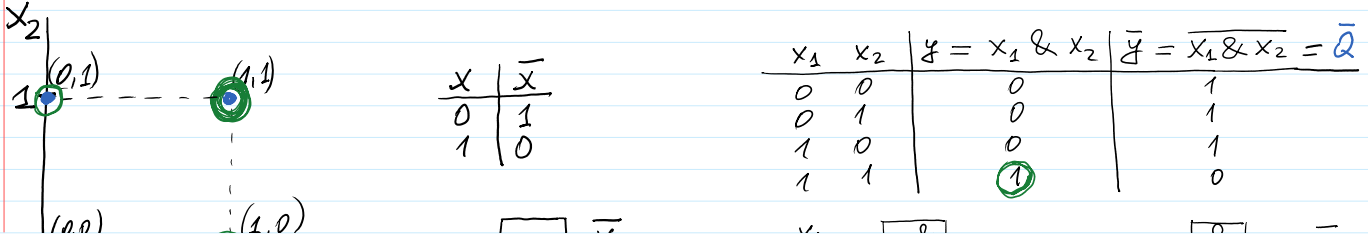
Our SRAM PUF-based security solutions are very suitable for applications such as secure key generation and storage, device authentication, flexible key provisioning and chip asset management. They can be used to secure payments, to protect highly sensitive data, for anti-counterfeiting and anti-cloning, to prevent identity theft, piracy of media content and software apps, software reverse engineering, and more.

Intrinsic ID's security solutions are available as hard and soft Intellectual Property (IP) and are used by companies who want a proven, easy and cost-efficient way to provide a solid trust base within their devices and applications.

From <https://www.intrinsic-id.com/sram-puf-technology-solutions/>

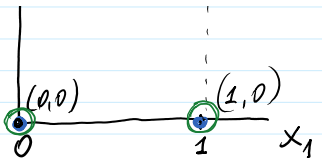
Principle of work with logical gates

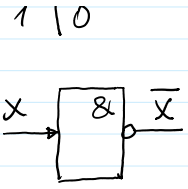
Logical Inversion, AND, NAND : $x \in \{0,1\}$; $x_1, x_2 \in \{0,1\}^2$

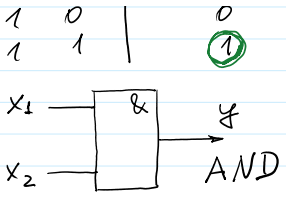


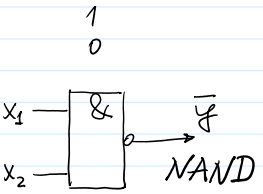
x	$\bar{x}$
0	1
1	0

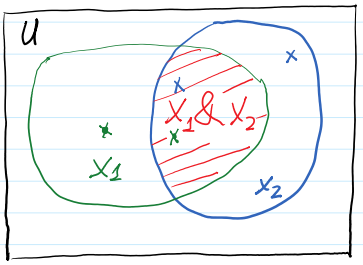
x_1	x_2	$y = x_1 \& x_2$	$\bar{y} = \overline{x_1 \& x_2} = \bar{Q}$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

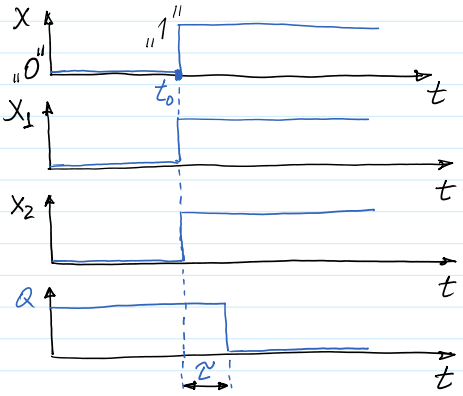


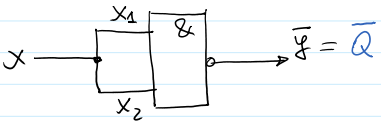








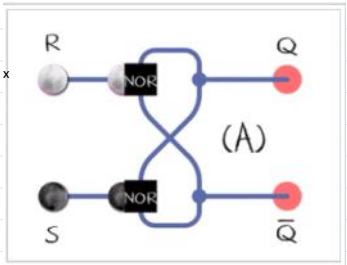




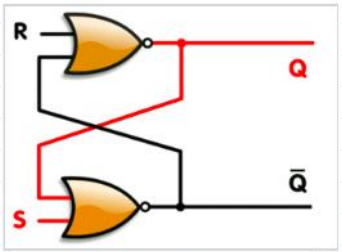
x_1	x_2	x	$\bar{y} = \bar{Q}$
0	0	0	1
1	1	1	0

Flip-flop (electronics)

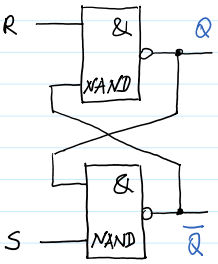
From <https://en.wikipedia.org/wiki/Flip-flop_(electronics)>



An animated SR latch. Black and white mean logical '1' and '0', respectively.
(A) S = 1, R = 0: set
(B) S = 0, R = 0: hold
(C) S = 0, R = 1: reset
(D) S = 1, R = 1: not allowed
The restricted combination (D) leads to an unstable state.



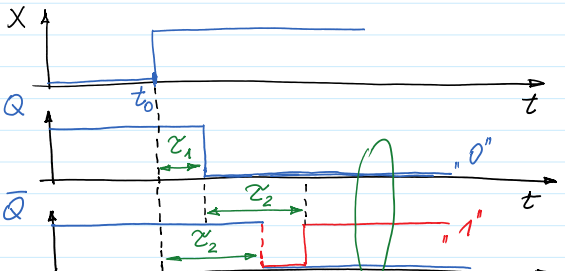
An animation of a SR latch, constructed from a pair of cross-coupled NOR gates. Red and black mean logical '1' and '0', respectively.

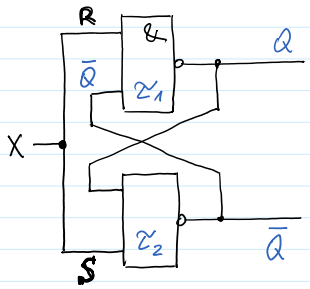


R	S	Q	$\bar{Q}$
1	0	0	1
1	1	0	1
0	1	1	0
1	1	1	0
0	0	1	1
1	1	?	?

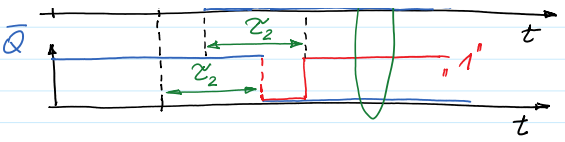
RS-Flip-flop

If $\tau_1 < \tau_2$

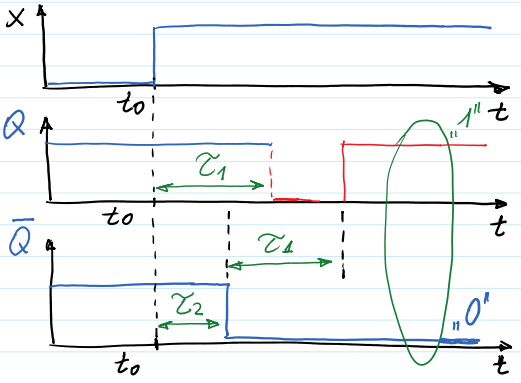


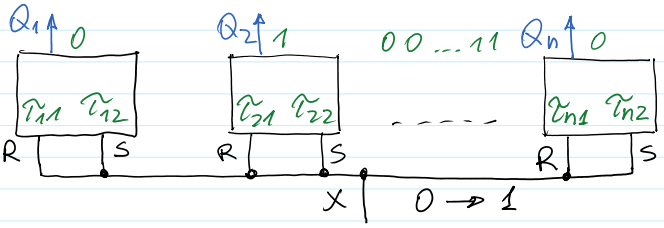


If $\tau_1 < \tau_2$



If $\tau_1 > \tau_2$



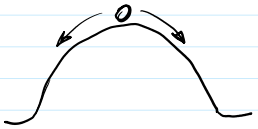


(0100 .... 11 0)

1 cipas su n RS-trig.

$\tau_{21} \approx \tau_{22}$ $\quad \tau_{i1} \approx \tau_{i2}$ $\quad \tau_{j1} \approx \tau_{j2}$

$\tau_{21} < \tau_{22} \rightarrow \tau_{21} > \tau_{22}$ $\quad \tau_{i1} > \tau_{i2} \rightarrow \tau_{i1} < \tau_{i2}$ $\quad$ - - - -



About 15% of RS flip-flops are unstable due to $\tau_{i1} \approx \tau_{i2}$ and are sensible to random changing of environment conditions.
Every transition of X from 0 → 1 will have 15% errors
Solution is to apply Error Correcting Codes (ECC):
to correct ~ 25% errors.
It is recomended to use 320 – 400 bits PUF.
Using 400 bits PUF we have 25% of correct bits values, i.e. 300 bits.
How many PUFs can be produced and distributed?
different
$N_{PUFs} = 2^{300} \approx 10^{100}$; The number of Planet population will be soon 8 Mlrd $= 8 \cdot 10^9$.