

# DNA based in silicon (Inchrosil)

Carlos Llopis, Silvia Llopis, Jose Daniel Llopis

**Abstract**— From Watson-Crick discovered DNA structure; there are lot researches in several fields of science about this topic, in special biology. Computer science is not an exception, Professor Adleman with his research and experimentation about new computation with DNA, opened new research way with organic DNA strands (which is called DNA Computing). In this article, we would like to give another point view, with artificial materials, by this reason, the document relates to an electronic system for emulating the DNA strand of a chromosome, which is characterized in that it includes means for the binary coding of the four types of nucleotides (A, G, C, T) that form the strands, such that the nucleotides that form complementary links are assigned complementary codes. Furthermore, with one minimal electronic module to emulate pair of nucleotides, it is possible to create complex electronic structures to solve NP and NP-Complete problems in polynomial times.

**Index Terms**— DNA, Computing DNA, Silicon, nucleotide, double helix, silicon, numerical systems.

## 1 INTRODUCTION

In 1953, Watson and Crick described structure of DNA [12], this fact is a revolution in the science, and consequently, several science areas use DNA structure for their studies. In special, Computer Science and Mathematics have used this simple structure of DNA to create a complex paradigms or methodologies to calculate unsolvable problem (Hamiltonian path problem, SAT, etc.). The new branch of science is called DNA Computing, which is a form of computing, using DNA strands and biochemistry methods. DNA computing is inside of Natural Computing [6], is a fast developing interdisciplinary area. Big hit in this area is in 1994, when Professor Adleman at University of Southern of California demonstrated using organic DNA can solve seven-point Hamiltonian problem [1] in polynomial time.

The Adleman experiments demonstrated to have more advantages of actual machines, based in Von Neumann architecture (Von Neumann, 1945) and Harvard Architecture [5]. Consequently of initial Adleman experiment, several searchers have developed machines, using Turing machine models [10] or similar machines with same power, but using DNA [3], [7].

In 1997, Mitsunori Ogihara and Animesh Ray described an implementation about Boolean Circuits using organic DNA [9]. Later in 2002 at Israel, Professor Ehud Shapiro Yaakov Benenson, Binyamin Gil, Uri Ben-Dor, and Rivka Adar had constructed a DNA device coupled with an input and output module, which would theoretically be capable of diagnosing cancerous activity within a cell, and releasing an anti-cancer drug upon diagnosis. [2].

In general, DNA computing is similar to parallel computing, because it's possible to solve any logical-mathematical problem in linear time. This is because; DNA structure consists in two chains of nucleotides (main and its complementary), and in nature both chain is linked in its nucleotides by hydrogen links. Sometimes by artificial or natural way, one DNA chain is not linked with complementary (In future, this chain could be called incomplete chain), first action of this incomplete chain is to look for its complementary chain, because its natural state of DNA strand. If in specific moment, there are lot incomplete chains, first action of theses chains is assembling with their complementary at the same

time, due massive parallelism inherent in DNA reactions, by this reason, if one mathematical problem is codified with incomplete chains (inputs, variables and equations), when all these chains are introduced in same place react creating a full strand, which is final result of mathematical problem. Consequently, complex problems as NP and NP-complete could be solved in linear times, using this procedure.

Note, all methodologies are based in organic DNA and chemical procedures, thus high knowledge in mathematics. Main idea of DNA Computing to be strong alternative of silicon computing (traditional computers), but DNA computing has several walls to cross, material used dies over time, because it's organic and perishable and it doesn't work at industrial environments, by this reason, it is very difficult to move outside of laboratories. Almost, there are several organic devices, which are working as a silicon computer, but only in scientific environments.

This article describes an electronic system emulate DNA structure without chemical reactions, using artificial components as silicon, creating with it an inorganic DNA with electronic circuits, by this reason, the electronic system is called InChroSil (**In**organic **CHRO**mosome based in **SIL**icon).

On the other hand, the system can be reused lot times, because it is composed by electronic circuits, there are not ethical or moral implications with cloning DNA (note, some countries are prohibited to manipulate human DNA). Also, this computation is not a pure DNA Computing; it is a mixture between both worlds (silicon and DNA), by this reason, it takes the best parts of both fields.

## 2 NUCLEOTIDE CODE AND STRAND STRUCTURE

To understand the electronic system, it is necessary to make a genetic introduction of chromosomes, which are formed by DNA, RNA and proteins. In turn, DNA is composed of two longitudinal fibres joined by centromere. These fibres are called chromatids and represent two identical chains of the duplicated DNA.

Therefore, DNA is composed of chains (set of nucleotides), which are differentiated in four types, which are listed below:

- Adenine is represented by the letter A.
- Guanine is represented by the letter G.
- Cytosine is represented by the letter C.
- Thymine is represented by the letter T.

Consequently, DNA is composed of two linked chains and each chain contains innumerable nucleotides. There are no complementary nucleotides rules on the same chain, i.e. they can join without any restriction. However, on linking the two chains, there are restrictions of complementarity, so that an Adenine (A) of a single chain can only join with Thymines (T) of the complementary chain, in the same way as Guanines (G) can only link with Cytosines (C). All above descriptions in this section are obvious and incomplete for any Biotechnical, geneticist or similar technical professional, but it's necessary for other readers to take a general idea.

On the other hand, emulating to nature, the electronic system is characterised in that it comprises means of binary encoding of the four different types of nucleotides which compose the strands, so that the nucleotides which form complementary linkages are assigned complementary codes by the inversion of one of them.

Binary encoding consists in two bits corresponding to each nucleotide acquire the decimal quantities of 0 of 3 so that it permits forming a plurality of numerical presentations determined by the position of the two bits and by the value each position receives. This encoding permits that any encoding system which has correspondence with the decimal or binary can be encoded with the numeration and encoding system of this electronic system. Next list shows codification for each nucleotide:

- Adenine (A) is assigned the value of "00" or in decimal 0.
- Guanine (G) is assigned the value of "01" or in decimal 1.
- Cytosine (C) is assigned the value of "10" or in decimal 2.
- Thymine (T) is assigned the value of "11" or in decimal 3.

Electronic schema (fig. 1) consists mainly in four bi-states and logical gates. Note one bi-stable can be used to store state information, by this reason, each bi-stable store one bit of nucleotide codified. Also, the two bi-stables connect with SEL0 signal represent one nucleotide and the two bi-stables connect with SEL1 signal represents codification of both nucleotides, so this codification permits to obtain different structures of both nucleotides, next list shows all possible combinations:

- Both nucleotides are active (state normal in nature).
- Top nucleotide is missing, hole in top of DNA strand.
- Bottom nucleotide (complementary) is missing, hole in bottom of DNA strand.
- Both are missing, not genetic information in place, break point of DNA strand.

All combinations are necessary to do holes in DNA strands, this skill very important to do DNA computing, because one mathe-

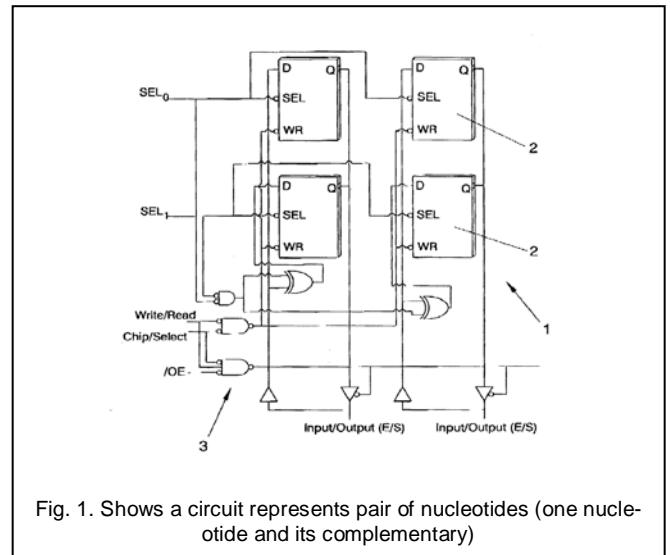


Fig. 1. Shows a circuit represents pair of nucleotides (one nucleotide and its complementary)

matical-logical problem could be represented by incomplete chain or chain with holes.

Furthermore, the system can read/write one specific nucleotide or both, depending of R/W signal and Chip/Select signal. Also, Fig. 1 is a conceptual model, by this reason; it could be developed in several technologies as CMOS, quantum or future technologies.

So the above electronic system is the unit to represent two nucleotides of DNA strand (one nucleotide and its complementary). Connecting these units in serial, it is possible to obtain full DNA strands, because there are not restrictions to join between nucleotides in same level, by this reason, the length of DNA strand depends only of number of units connected, creating with this way, a three-dimensional DNA database with different access it (by columns, rows and one simple nucleotide).

## 2 NUMERICAL SYSTEM AND ARITHMETIC OPERATION

Numerical System is essential to write specific symbol as number and to create complex mathematical equations or formulas. In this article describes numerical system in base 4, which uses the nucleotides of DNA strand (A, G, C and T) as symbols. Conventionally, the encoding of organic DNA is known macroscopically, i.e. encoding a chain or portion, or also encoding the operations with its operands [4]. These encodings do not permit as much flexibility because they limit the number of combinations that can be performed, by this reason, numerical system described below consists of a system where genetic information is encoded microscopically, i.e. using nucleotides codes. This characteristic of atomicity permits performing combinations to generate a numerical system, which has a potency and scalability, in the representation of numerical quantities.

Nucleotide code is defined as two bits sequence, which acquires value from zero to three. Also any chain of nucleotide

represents a numerical quantity, because, this numerical system is positional and any symbol (nucleotide) inside of chain has two meanings; first, position within the numerical chain, and second, value of symbol receives (from 0 to 3). With both data, it is possible to obtain partially final value of numerical quantity. Fig 2 shows the correspondence between numerical system described and other systems.

Decimal	Binary	Octal	Hexadecimal	InChro Sil
0	00	0	0	A
1	01	1	1	G
2	10	2	2	C
3	11	3	3	T
4	100	4	4	GA
5	101	5	5	GG
6	110	6	6	GC
7	111	7	7	GT
8	1000	10	A	CA
9	1001	11	B	CG
10	1010	12	G	CC
11	1011	13	D	CT
...	...	...	...	...

Fig. 2. Correspondence between numerical system and others systems.

An example is always more clarifying than a set of words, by this reason, if we consider number 4567<sub>10</sub> in decimal base, the corresponding binary representation is 000100011101011<sub>2</sub>, in octal is 010727<sub>8</sub>, in hexadecimal is 11D7<sub>16</sub> and finally, in this numerical system (In a future could be Cod-Inchrosil) encoding it is AGAGTGGT<sub>4</sub>. It is also possible to change from a numerical system using a next polynomial equation [8]:

$$\text{Number } (X_{n-1} \dots X_2X_1X_0)_b = X_{n-1}b^{n-1} + \dots + X_2b^2 + X_1b^1 + X_0b^0$$

Where  $X_{n-1} \dots X_2X_1X_0$  are digits of number and  $b$  base used in codification. For example, next number encoded in Cod-Inchrosil: AGAGTGGT<sub>4</sub> can convert in decimal:

$$AGAGTGGT_4 = A \cdot 4^7 + G \cdot 4^6 + A \cdot 4^5 + G \cdot 4^4 + T \cdot 4^3 + G \cdot 4^2 + G \cdot 4^1 + T \cdot 4^0 = 0 \cdot 4^7 + 1 \cdot 4^6 + 0 \cdot 4^5 + 1 \cdot 4^4 + 3 \cdot 4^3 + 1 \cdot 4^2 + 1 \cdot 4^1 + 3 \cdot 4^0 = 4567_{10}$$

On the other hand, inverse operation from decimal number to Cod-Inchrosil number is as follows:

Supposing decimal number to be 189<sub>10</sub>

$$\begin{aligned} 189 \div 4 &= 47 \rightarrow \text{minus } (189.4) = 1 \rightarrow G \\ 47 \div 4 &= 11 \rightarrow \text{minus } (47.4) = 3 \rightarrow T \\ 11 \div 4 &= 2 \rightarrow \text{minus } (11.4) = 3 \rightarrow T \\ 2 \div 4 &= 2 \rightarrow \text{minus } (2.4) = 2 \rightarrow C \end{aligned}$$

Final result - 189<sub>10</sub> = CTTG<sub>4</sub>

Furthermore, with decimal numbers as 873.875<sub>10</sub> to convert DNA codification, Integer part is calculated with method above, being TGCCG<sub>4</sub>. Also decimal part is calculated as follows:

$$\begin{aligned} 0.875 \times 4 &= 3.5 \rightarrow \text{the integer is 3 so is T.} \\ 0.5 \times 4 &= 2 \rightarrow \text{the integer is 2 so is C.} \end{aligned}$$

Final number is TGCCG.TC<sub>4</sub>.

In general, numerical system can represent any integer and decimal representation, being compatible with other numerical systems.

Also, this article describes a several arithmetic operation with numerical quantities (addition, subtraction, multiplication, division, etc.). In particular about addition operation is gives the guidelines for the addition of two nucleotides

	A	G	C	T
A	A	G	C	T
G	G	C	T	GA
C	C	T	GA	GG
T	T	GA	GG	GC

Fig. 3. Guidelines for the addition of two nucleotides

For example, suggesting two numbers ATCGATA and TAGCCAA, the addition both number as follow:

	0	Carry
	ATCGATA	Number 1 (3660 in decimal)
+	TAGCCAA	Number 2 (12704 in decimal)
	TTTTCTA	Result (16364 in decimal)

Also other example with carry in addition operation as follow:

	G G G G G	Carry
	A T C G A T A	Number 1 (3660 in decimal)
+	A T C T T T C	Number 2 (3838 in decimal)
	G ↑ T ↑ G ↑ G ↑ A ↑ C C	Result (7498 in decimal or GTGGACC)

Next operation to define is the subtraction, which is known as an

	A	G	C	T
A	A	GT	GC	GG
G	G	A	GT	GC
C	C	G	A	GT
T	T	C	G	A

Fig. 4. Guidelines for the subtraction of two nucleotides

arithmetic operation opposite to addition. The guidelines to subtract two nucleotides are defined following Fig 4.

Good way to describe the subtraction is by means of one example. Considering, two numbers in Cod-Inchrosil: TGAT<sub>4</sub> (in decimal 211) and GCCC<sub>4</sub> (in decimal 106), the result of the subtraction is 105 in decimal and GCCG<sub>4</sub> in Cod-Inchrosil, then the calculation is performed with the subtraction that has been showed in next operation Fig 5.

	T	G	A	T	number 1
-	G+G=C	C+G=T	C	C	number 2
	G	\ C	\ C	G	result

Fig. 5. Example of subtraction of two numbers

How show in operation above (Fig 5), when T and C is subtracted no carry occurs, and the result is G, on the other hand, when A and C is subtracted, the result is C, but the carry is G, this quantity has to be added to the following nucleotide, in this case it is C. T is obtained from this addition operation. In general, the mechanism is the same as in the subtraction in decimal, but in this case is used nucleotides. Other way to do subtraction operation is by means of the addition first operand and two's complement of second operand. That is very easy to do, because it is possible to obtain complementary chain, and add G to obtain two's complement.

The following operation defined is multiplication. The multiplication or product is defined as an arithmetic operation where successive additions are performed. As with previous operations, guidelines are given for multiplication at a nucleotide level, these products are observed following Fig 6:

	A	G	C	T
A	AA	AA	AA	AA
G	AA	AG	AC	AT
C	AA	AC	GA	GC
T	AA	AT	GC	CG

Fig. 6. Multiplication between two nucleotides

Considering two numbers, the multiplication of both numbers is showed in Fig 7:

	GTTG	number 1
x	GGA	number 2
	AA	product G X A
	AAA	product T X A(*)
	AAAA	product T X A(*)
+	AAAAA	product G X A(*)
	AAAAA	Partial sum
	AGA	product G X G
	ATAA	product T X G(*)
	ATAAA	product T X G(*)
+	AGAAAA	product G X G(*)
	AGTTGA	Partial sum
	AGAA	product G X G
	ATAAA	product T X G(*)
	ATAAAA	product T X G(*)
+	AGAAAAA	product G X G(*)
	ACGTAGA	Final result (2500 in decimal)

(\*) A's area added on the right

Fig. 7. Example multiplication.

With this numerical system could be represented decimal

numbers, note, all floating point numbers (decimal numbers) follow an established pattern, which can be summarized in the following equation:  $r = m \cdot b^e$ , where r is the real number, m is the mantissa, b is the base and, finally, e is the exponent, which permits raising the base. For example, number 0.00012310 in fixed point (we use the decimal system), in floating point could have the following representations:  $1.23 \cdot 10^{-4}$ ,  $0.123 \cdot 10^{-3}$ ,  $0.0123 \cdot 10^{-2}$ ,  $0.00123 \cdot 10^{-1}$ , etc. As is observed in the previous representations, all make reference to the same number but with different representation in floating point, in DNA codification is not an exception about ambiguous representation.

By this reason, and to avoid ambiguities in the representations of numerical quantities, a standard (IEEE, 1985) was established which permits the homogenous representation of floating point numbers. Consequently, this numerical system has adapted this standard to DNA codification. Next tables show different representation in float point of Cod-Inchrosil, where each cell in Fig 8 and Fig 9 represents nucleotides.

Sign	Strand	Cod	Exp	Hebra	Cod	Mantiss
1	1	1	4	1	1	11

Fig. 8. Simple Precision

Sign	Strand	Cod	Exp	Strand	Cod	Mantiss
1	1	1	7	1	1	28

Fig. 9. Double Precision

Codifications are represented in Fig. 10:

Strand	Fixed point + sign	Complement A1	Complement A2	Excess 4 <sup>4</sup>
Strand1	GTTT (+127 <sub>10</sub> )	GTTT (+127 <sub>10</sub> )	GTTT (+127 <sub>10</sub> )	TTTT (+127 <sub>10</sub> )
Strand2	CAAA (+0 <sub>10</sub> )	CAAA (-127 <sub>10</sub> )	CAAA (-128 <sub>10</sub> )	AAAA (-128 <sub>10</sub> )
Strand1	...	...	...	...
Strand2	...	...	...	...
Strand1	AAAA (+0 <sub>10</sub> )	AAAA (+0 <sub>10</sub> )	AAAA (+0 <sub>10</sub> )	CAAA (+0 <sub>10</sub> )
Strand2	TTTT (-127 <sub>10</sub> )	TTTT (-0 <sub>10</sub> )	TTTT (-1 <sub>10</sub> )	GTTT (-1 <sub>10</sub> )
Strand1	CAAA (-0 <sub>10</sub> )	TTTT (-0 <sub>10</sub> )	AAAA (-0 <sub>10</sub> )	CAAA (+0 <sub>10</sub> )
Strand2	GTTT (+127 <sub>10</sub> )	AAAA (+0 <sub>10</sub> )	TTTT (-1 <sub>10</sub> )	GTTT (-1 <sub>10</sub> )
Strand1	...	...	...	...
Strand2	...	...	...	...
Strand1	TTTT (-127 <sub>10</sub> )	CAAA (-127 <sub>10</sub> )	CAAG (-127 <sub>10</sub> )	AAAG (-127 <sub>10</sub> )
Strand2	AAAA (+0 <sub>10</sub> )	GTTT (+127 <sub>10</sub> )	GTTC (+126 <sub>10</sub> )	TTTC (+126 <sub>10</sub> )
Strand1	...	...	CAAA (-128 <sub>10</sub> )	AAAA (-128 <sub>10</sub> )
Strand2	...	...	GTTT (+127 <sub>10</sub> )	TTTT (+127 <sub>10</sub> )

Fig. 10. Several Codifications

## CONCLUSION

This article shows minimal unit of one artificial/electronic DNA, which has the same organization and structure of organic DNA (double helix) except for the chemical reactions, this device was developed using the standard technology Complementary metal-oxide-semiconductor (CMOS) and will be made in the nano-metric scale, being the first inorganic DNA based on silicon. Our future research, we would like to mix all theories of DNA computing with silicon components, because at the moment both are enemies. Other way, this system is perfect to do genetic experiment without ethic clonotations, because some countries are prohibited to manipulate organic material.

Inchrosil is used mostly for mass storage sequences DNA, with their uses:

- Fingerprinting staff, with the storage of the genetic fingerprint. His invention is Dr. Alec Jeffreys at the University of Leicester in 1984[14].
- Genetic studies, with the tool where large sequences would be stored, which could be compared and manipulated digitally.
- Genebanks, which store large amounts of genetic information.
- Classification of species and animals.
- Tool medical studies (genetic).
- Genetic computers.

InChroSil allows us to store in less space, more genetic information (saving 88 %). Current systems have a problem with the amount of information storage that organizations and institutions require. I.e. the human genome has approximately 3,000 million pairs of nucleotides, and is divided into 23 chromosomes. This information is too big for available systems, so currently only the principal chain is stored, meaning that holes within the DNA are not represented. InChroSil offers less space and more information, with higher performance. It can store the entire structure of the DNA (principal and secondary chains), meaning InChroSil can store and represent even incomplete chains including the existent holes within the DNA chain.

This device could use non-volatile digital memory (as we said before), less storage space, is portable and allows for rewriting. It is also able to store non-genetic information in the nucleotides (the basic elements of the DNA strand). The nucleotides can be used to represent genetic information, text, files, or even pictures in a more compact manner and with a wide range of device, hardware and software compatibility.

## FUTURE RESEARCH

Our future works are focused in computational systems using this technology, because it is possible to solve several NP-problems with mathematical knowledges of DNA computing, or to create new methodologies using Inchrosil.

## REFERENCE

- [1]. Adleman, L. M. (1994). Molecular computation of solutions to combinatorial problems. *Science*, 1021–1024.
- [2]. Benenson, Y., Gil, B., Ben-Dor, U., Adar, R., & Shapiro, E. (2004). An autonomous molecular computer for logical control of gene expression. *Nature*, 423–429.
- [3]. Boneh, D., Dunworth, C., Lipton, R. J., & Sgall, J. I. (1996). On the computational power of DNA. *Discrete Applied Mathematics*, 79–94.
- [4]. F, G., M, F., & C, B. (1996). Making DNA add. *Science*, 220–223.
- [5]. Furber, S. B. (1989). *VLSI Risc Architecture and Organization*. CRC Press.
- [6]. G.Rozenberg, T.Back, & J.Kok. (2012). *Handbook of Natural Computing*. Springer.
- [7]. Kari, L., Gloor, G., & Yu, S. (2000). Using DNA to solve the Bounded Post Correspondence Problem. *Theoretical Computer Science*, 192–203.
- [8]. Knuth, D. (1962). Positional Number Systems. In *The Art of Computer Programming* (pp. 194–213). Addison–Wesley.
- [9]. Ogihara, M., & Ray, A. (1999). Simulating Boolean circuits on a DNA computer. *Algorithmica*, 239–250.
- [10]. Turing, A. (1936). On Computable Numbers, with an Application to the Entscheidungsproblem., (pp. 230–65).
- [11]. Von Neumann, J. (1945). First Draft of a Report on the EDVAC.
- [12]. Watson, J., & Crick, F. (1953). Molecular structure of nucleic acids; a structure for deoxyribose nucleic acid. *Nature*, 737–738.
- [13]. Wu, G., & Seeman, N. C. (2006). Multiplying with DNA. *Natural Computing*, 427–441.
- [14]. Jeffreys A.J., Wilson V., Thein S.W. (1984). "Hypervariable 'minisatellite' regions in human DNA". *Nature* 314: 67–73.