Digital DNA Molecules

Hilmar Rauhe[*], Gaby Vopper[&], Udo Feldkamp[*], Wolfgang Banzhaf[*], Jonathan C. Howard[&]

February 11, 2000

rauhe@LS11.cs.uni-dortmund.de
feldkamp@LS11.cs.uni-dortmund.de
banzhaf@LS11.cs.uni-dortmund.de
jonathan.howard@uni-koeln.de

Abstract:

An approach based on programmable self-assembly of DNA oligonucleotides was used to create digital DNA molecules representing binary datastructures which are equivalent to those used in computers. Utilizing plasmids as a kind of computer memory, the digital molecules could be isolated, amplified and read out using common genetic techniques. Programmability allowed several applications to be realized *in vitro* such as a fast physical random number generator and digital DNA-"barcodes".

[*] Dept. of Computer Science, LS11, University of Dortmund, Joseph-von-Fraunhofer Str. 20, D-44227 Dortmund, Germany

[&] Institute for Genetics, University of Cologne, D-50674 Cologne, Germany

An approach to DNA computing[1] is shown which is based on programmable, linear self-assembly[2, 3]. It is focused on digital molecules that are compatible to binary data used in computers and to biological structures. The compatibility to biological structures in particular enabled the digital molecules to be generated, isolated, amplified and read out using only standard lab-techniques. The fact that grammars[4, 5] provide programmability by an ordered, rule-based construction of specific structures which can be concatenations of symbols as well as concatenations of molecules allowed several applications of the digital molecules (see below).

For implementation of digital DNA molecules we used the grammar G_{rand} = (Σ, V, R, S) with Σ:= {0, 1, s, e}, V:= {A}, S and R:={S:=sA, A→0A, A→1A, A→e}, where 0, 1 represent bits and s, e terminators (start, end) (see Figure 1). G_{rand} describes the production of digital strings beginning with s and ending with e and an arbitrary number of bits in between (see Figure 2). Reading the generated strings as binary representations of numbers the grammar describes a random number generator utilizing the principle that the application of A→0A or A→1A *in vitro* is random.

The rules of the grammar G_{rand} were implemented by "rule-molecules"[2] referred to as "algomers", short double stranded DNA molecules with sticky ends (see Figure 1 and Figure 2). Each algomer corresponded to exactly one rule of the grammar. Algomers consisted of two annealed complementary oligonucleotides and were constructed such that their double stranded core sequences represented terminals and their sticky ends represented variables (see Figure 1).

```
S:=sA                                  _
    H              s                   A
5′ agctt tatatctccatttgccctagtgaag          3′
3′     a atatagaggtaaacgggatcacttg   aacc  5′

A->0A                                  _
    A              0                   A
5′  ttgg cgagatatcaacgccacccttgctt          3′
3′       gctctatagttgcggtggggaacgaa   aacc  5′

A->1A                                  _
    A              1                   A
5′  ttgg cgacccgggaaacaactattgctagt         3′
3′       gctgggccctttgttgataacgatca   aacc  5′

A->e                                   _
    A              e                   B
5′  ttgg tgcgggagttggaagcaactacgat   g     3′
3′       acgccctcaaccttcgttgatgcta   cctag 5′
```

Figure 1**:** DNA molecules that were used for implementation of the rules of G_{rand}. Every molecule represents one rule. A→0A and A→1A can concatenate in either direction. S:=sA (start) and A→e (end) work as terminators and can concatenate only right or left-handed. The double stranded core sequences s, e, 0, 1 represent terminals, the sticky end A represents a variable. H and B on terminators are overhangs compatible to restriction sites (H = HindIII, B = BamHI) and were added for subsequent plasmid cloning. All molecules ligate in defined manner and direction.

In the case of bits the oligonucleotides were constructed as "elongators", molecules that can anneal in either direction with defined orientation, and were phosphorylated for subsequent ligation. The start and end algomers were designed such that they anneal with one end to the bits and with the other to a cloning site (start: HindIII, end: BamHI).

Assembly of each algomer was then done by mixing the corresponding oligonucleotides and annealing them under controlled conditions (see Materials and methods).

Once assembled, the algomers were concatenated by ligation of their sticky ends, thus applying the rules of G_{rand}. Molecules yielded from this reaction are digital DNA polymers of bits and are terminated by start and end sites (see Figure 2 and Figure 3). They represent words of the language $L(G_{rand})$ (thus called "logomers") and are the result of the algorithm defined by G_{rand}. After complete ligation, the resulting logomers could be cloned in order to isolate and amplify single molecules (see Figure 2).

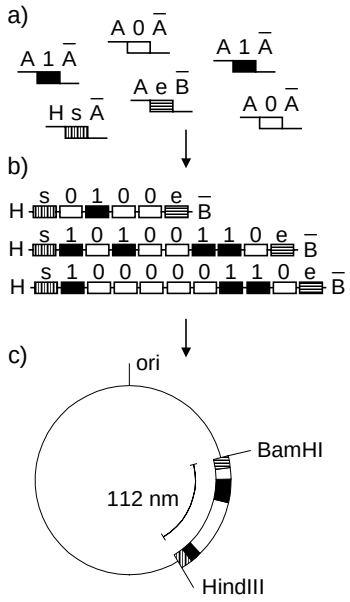

Figure 2: a) Molecules representing the rules of G_{rand} ("algomers"). b) Concatemers ("logomers") yielded by self-assembly. All logomers are of the form s{0|1}e, consisting of a left-handed bracket s, a right-handed bracket e and an arbitrary number of bits in between. The length of the logomers in terms of algomers can be controlled stochastically by the ratio of elongators to terminators or exactly by the grammar itself if rules are set up to generate polymers of defined size (see text). c) Isolation of single logomers by cloning. The digital "printout" of the cloned logomer can be seen in Figure 4.

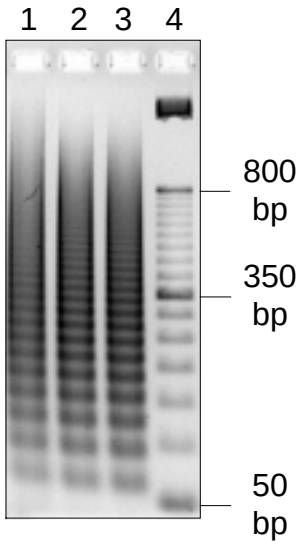

Figure 3: Result of the self-assembly of the algomers. The bands show a discrete pattern of logomers of different lengths, each band consisting of a plurality of logomers of same length. The logomers sizes range from 0 bits (60bp) in steps of 30bp per bit to approximately 32 bits (1020bp). Lane 1 shows the result of ligation of 0s, Lane 2 result of ligation of 1s, Lane 3 result of a ligation of 0s and 1s. Lane 4 contains a 50bp molecular weight marker. Lanes 1 to 3 each contain about 10^{12} random logomers out of a total of 3×10^{13} generated during a single reaction.

After isolation, cloned logomers were read out by a digital DNA typing procedure originally developed for minisatellite analysis[6]. The method allowed readout of single logomers directly by PCR. The outputs were digital "printouts" of logomers that could be read by gel-electrophoresis (see Figure 4 and Materials and methods).

Using these methods G_{rand} was used for generation of physical random numbers (see Figure 3 and Figure 4). The performance of G_{rand} was close to that of pseudo random number generators on common hardware (see Figure 8).

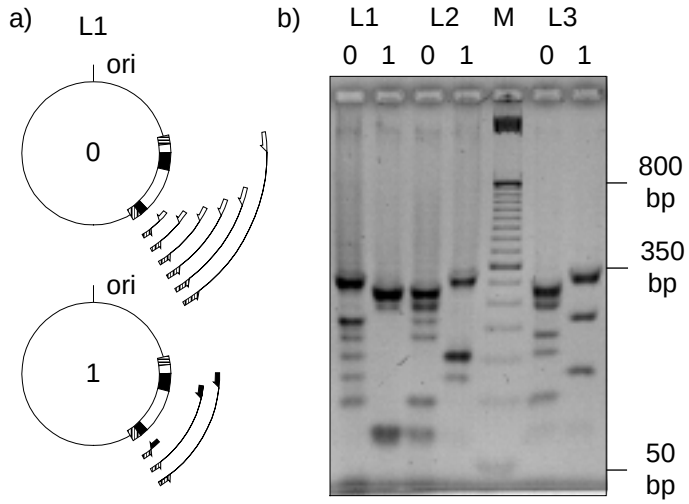

Figure 4**:** a) Schematic description of digital DNA Typing. Logomer L1 is read out using two PCR reactions. Each reaction contains the start-primer as first and either the 0- or the1-primer as second primer. The resulting complementary pattern can be seen in b). b) Digital "printouts" of three 9-bit random numbers visualized by gel-electrophoresis. Read bottom up, L1 equals 100000110_{bin} = 262_{dec}, L2 equals 001100001_{bin} = 97_{dec}, L3 equals 101001001_{bin} = 329_{dec}. M is a 50bp molecular weight marker.

Generalizing the method shown for G_{rand} a compiler was developed to translate regular grammars directly to DNA sequences[7]. It allows the implementation of arbitrary regular grammars. Under the conditions of our system the populations of logomers after ligation had about $3x10^{13}$ molecules, their sizes ranging up to 32 bits (see Figure 3). In order to process the result of the computation further the logomers were isolated by ligating them into plasmids and cloning them into bacteria (see Figure 2). Utilizing plasmids as a kind of computer memory to which data was written by cloning of logomers it became possible to perform parallel computation at the level of single molecules. The logomer-represented results of the computational operations could then be amplified to the macroscopic range by simply growing bacteria.

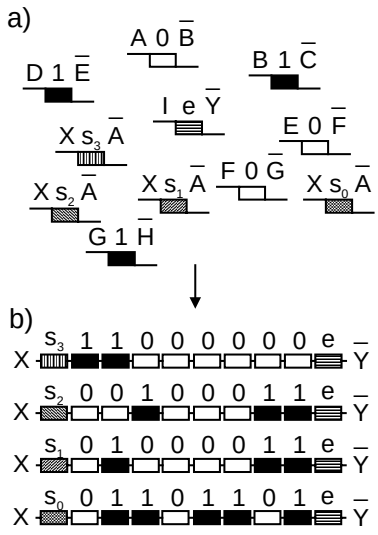

Figure 5: Creation of 32-bit datatypes in DNA. a) Algomers, representing bits (0 or 1) and start or end sequences. b) Four Logomers, representing the four bytes of a 32-bit datastructure. Every byte contains a start sequence that marks its byte position. For instance, the shown datastructure can be read as an IP-address (s_0 = rightmost byte).

More complex data structures can be implemented using the grammar G_{dat} = (Σ, V, R, S), with Σ:= {0, 1, e, s_0, s_1, s_2, …, s_{n-1}}, V:={ A, B, C, D, E, F, G, H, I}, start- symbol S and R:= {S:=s_iA, A→0B, A→1B, B→0C, B→1C, C→0D, C→1D, D→0E, D→1E, E→0F, E→1F, F→0G, F→1G, G→0H, G→1H, H→0I, H→1I, I→e with i = {0, ..., n-1}}. The grammar describes representations of binary data as concatenation of 8 random bits (1 byte). Every byte carries a start sequence s_i which acts as a template of the forward primer during readout PCR. It can also be used to represent the position of a certain byte within a larger datastructure such as a 4-byte (32-bit) number or a string of bytes. A 1-byte grammar (n = 1) is sufficient to map datastructures like the whole ASCII alphabet to DNA sequences. A 4-byte grammar (see Figure 5 and Figure 6) is capable of mapping datastructures commonly used in computers like integers, floating point numbers or IP-addresses to DNA sequences. A n-byte grammar can be used for representation of strings of length n.

An application of these data structures are digital DNA-"barcodes" for the purpose of labelling. We tested several materials for their ability to be labelled by DNA, fluids such as motor-oil and paint and paper-based materials such as printer paper, 3M Post-Its and banknotes. Logomers were recovered from all tested materials and different concentrations of logomers used to check efficiency of recovery. In comparison to water, paint required 10 fold, oil 100 fold higher concentration (see Materials and methods). Labelling of paper-based materials was highly dependent on consistency of the used paper (see Figure 7).

Decisive in this context is the fact that DNA in itself is not toxic to biological organisms or to the environment, and sequences used for encoding can be constructed which do not contain any biologically relevant information. Since logomers can carry information such as product, manufacturer and expiration date they might be applicable for industrial purpose as well as for characterisation and identification of genetically engineered products such as food. Like the bacteria in our experiments which carry DNA numbers, genetically engineered products can be labelled genomically by recombination techniques[8]. In case of industrial applications our experiments suggest that small amounts of the labelled material should be sufficient for readout, for example from small amounts of car paint in case of an accident. If the start sequence is kept secret, digital DNA-Typing can directly be used for cryptography[9] requiring neither sequencing nor subcloning[10, 11]. In this case the readout procedure becomes a procedure of decryption as the forward readout primer works as a secret key. Readout without knowing the forward primer is prevented by adding excess dummy sequences, such as random DNA[9] or logomers with different priming sites[10, 11].

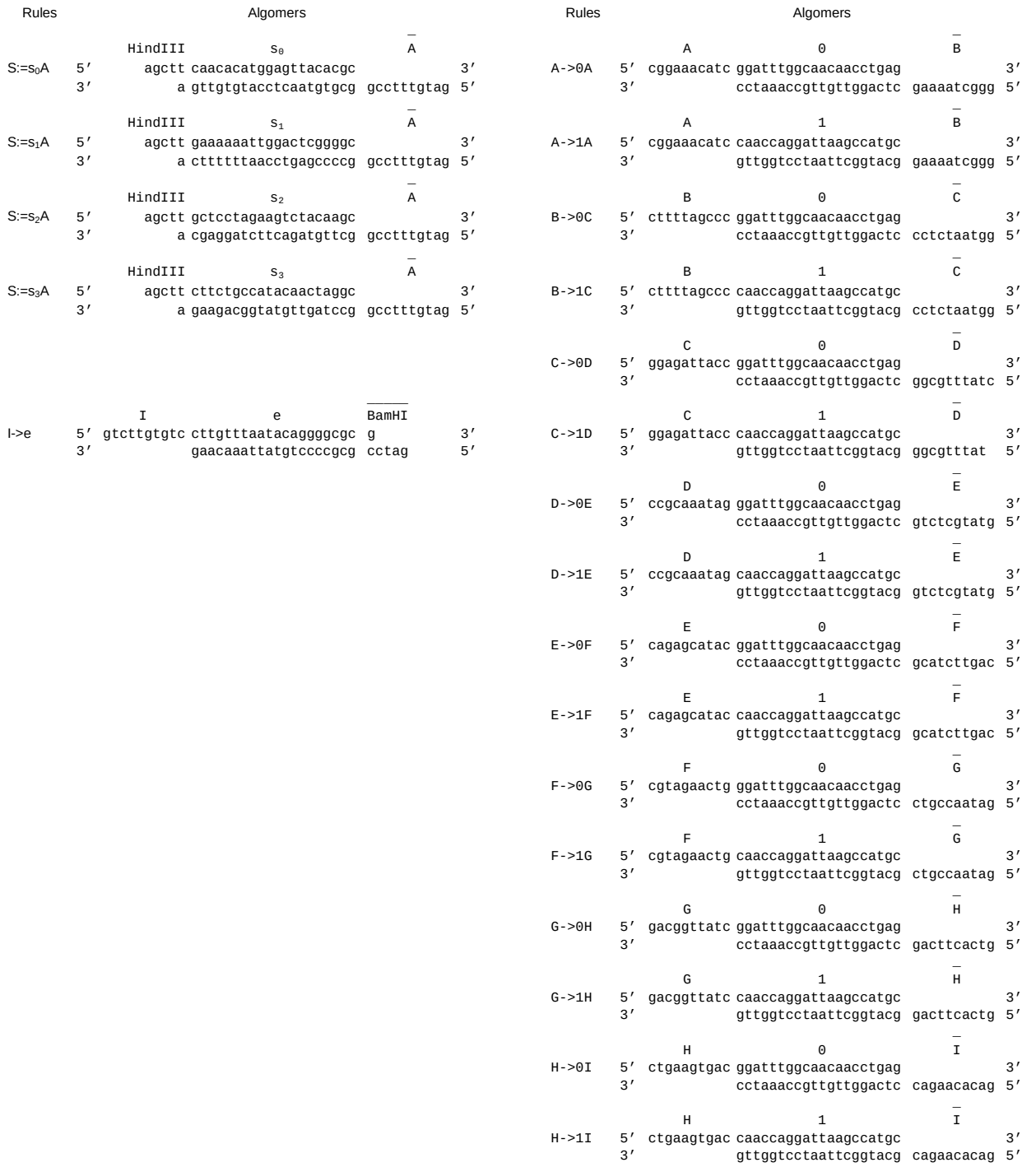

Figure 6: Rules of the 4-byte grammar and their translation to algomers as produced by the compiler. After ligation of the algomers, every digital molecule represents exactly one of 256 possible bytes and contains one of four possible start sequences which identify the byte position within a 32-bit datastructure. Thus the grammar is able to produce 2^{32} different items.

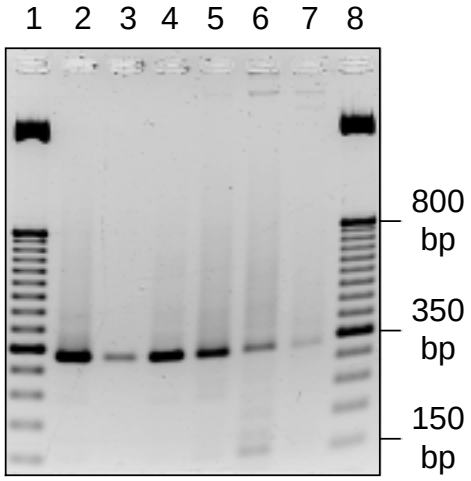

Figure 7: Labelling of various substances and materials. Lane 1 and 8 are a 50bp weight marker. Lanes 2-7 show PCR products of a 9-bit logomer after recovery from labelled materials. PCR was done with 5'-start and 3'-end primers to detect existence of logomer. Lane 2 shows readout after recovery from H_2O, Lane 3 shows readout after recovery from oil (HD 10W-40, TS-Union), Lane 4 after recovery from paint (Herbol Tafellack, Herbol), Lane 5 after recovery from Post-It (3M), Lane 6 after recovery from printer paper (80g/m², Produkta Etat), Lane 7 after recovery from banknote (10 DM note, Bundesdruckerei, Germany).

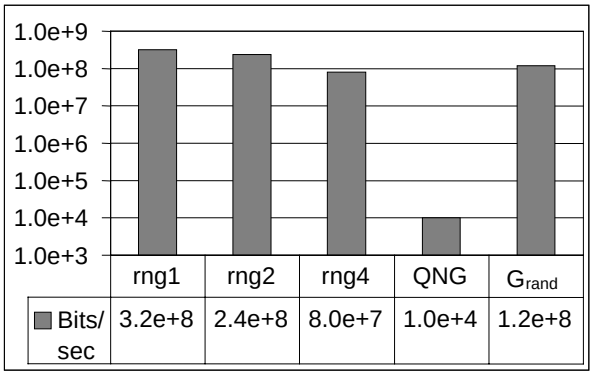

Figure 8: Benchmarking of different random number generators. rng1 – rng4 are pseudo-random number generators as described in Numerical Recipes[12] performed on a PC (AMD K6-II, 450 MHz). QNG is a commercial physical random number generator[13], its speed taken from its specification[13]. G_{rand} was performed as a 24h ligation using 1400 NEB units T4 DNA Ligase (see Materials and methods). Under these conditions G_{rand} has a theoretical maximum performance of 10^{13} bits per second based on the turnover of T4 DNA-Ligase, thus can be optimized further. Runtime-complexity of the physical random number generators is less than those of the silicon-based pseudo random number generators.

Discussion

An approach of programmable, linear self-assembly of DNA molecules was shown here. A digital structure of the molecules was chosen to represent binary datastructures that are "natural" to silicon computers while requiring only common lab techniques. The approach utilizes known techniques[1, 3, 6] towards further integration of a hybrid DNA-Silicon computing. Key features of the approach are its programmability and the procedure of readout. Both features can be used as interfaces of a hybrid DNA-Silicon architecture by accompanying the *in vitro* system with the compiler software and by reading the bits like nucleotides with an automated DNA sequencer. In such a hybrid architecture, programs are written on a computer, translated to DNA sequences by the compiler, synthesized by an oligo-synthesizer, performed *in vitro* by hybridization and ligation and the results being read back to the computer after PCR by a DNA sequencer. Another aspect of the compatibility to molecular genetics is that circular, plasmid-based DNA structures allow mass-production of nanocomponents[3, 14, 15, 16] that are otherwise practically impossible to synthesize on a large scale.

Although the approach has limitations, it is already capable of applications like the physical random number generator and DNA-"barcodes". DNA-barcodes were stable in and could be recovered from materials like paint and motoroil. By using techniques of gene targeting and recombination, DNA

barcodes can be used for labelling organisms and organic materials. Thus genetically engineered products can be labelled in order to identify them in subsequent products and food. Also, labelled organisms can be authenticated and monitored for the purpose of disease prevention which may help to avoid contamination of food. The DNA-barcodes can not only be used for authentication but also as arbitrary data. Moreover they can be encrypted[9], which might become of more than theoretical interest in case of an artificial "genetic" fingerprinting.

Linear self-assembly was supposed to be irrelevant for universal computation[2]. However, things are a bit different when constructing linear biological structures such as artificial chromosomes[17]. In that case terminals do not represent bits but biologically meaningful sequences such as restriction sites, recombination sites, promotors or whole genes. This might become more significant in the context of the idea of the programming of biological systems[18].

Materials and methods

Algomer assembly. Oligos were synthesized and PAGE-purified (ARK-Scientific, Darmstadt, Germany). All oligos were dissolved in H_2O 100µM and stored at –20°C. Elongators were phosphorylated, upper and lower strand seperately, in 20µl volume by adding 16µl DNA (100µM), 2µl ligation buffer (NEB) and 2µl PNK (Polynucleotide kinase, NEB) and incubating for 1 hour at 37°C. Algomers were assembled by mixing 20µl of upper and lower strand oligos (each 100µM) and annealing them in a thermocycler (PTC-100, MJ Research) 5' at 95°C and from 75°C to 50°C in steps of 1°C per minute.

Self-assembly. Terminators and elongators were mixed with a ratio of 1:8. In 27µl reaction volume 1µl start-algomer (50µM), 1µl end-algomer (50µM), 10µl 0-algomer (40µM) and 10µl 1-algomer (40µM), 1,5µl rATP (10mM) and 3,5µl T4 DNA Ligase (NEB) were mixed and incubated 2-3h at 22°C or 24h for benchmarking of the random number generators (Figure 8).

Isolation. The logomers yielded from self-assembly were cloned in pBluescriptIIKS+ (Stratagene). In 10µl reaction volume 5µl pBluescriptIIKS+ (5nM), 0,5µl mix of the self-assembly reaction, 1µl ligase buffer (NEB), 0,5µl T4 DNA Ligase (400 NEB units, NEB) and 3µl H_2O were mixed and incubated for 12h at 16°C. 5µl of the reaction mixed was used for subsequent transformation of e.coli DH5-α cells.

DNA typing. Cloned logomers were read out by a PCR-based digital DNA typing procedure originally developed for minisatellite analysis[6]. PCR was done either directly from a bacterial colony, by dilution of an overnight culture or from dilution of a plasmid preparation. Two PCR reactions were set up each containing the 5' start-primer and either the 3'-0-5' primer or the 3'-1-5' primer. The PCRs result in binary complementary ladder patterns of DNA fragments when visualized by gel-electrophoresis (4% Agarose, stained with 0.0005% Ethidiumbromide). Each PCR was prepared in 200µl reaction volume by mixing 144µl H_2O, 20µl PCR Buffer 10x (100 mM Tris-HCl, 500 mM KCl, pH 8.3 at 20°C), 20µl $MgCl_2$ (25mM), 4µl dNTPs (10mM, Pharmacia Biotech), 2µl Taq-Polymerase (5u/µl, Gibco-BRL), 4µl 5' Primer (10µM), 4µl 3' Primer (10µM) and 4µl logomer cloned in pBluescriptIIKS+ (10^6 molecules/µl) as template. PCR was performed in a thermocycler (PTC-100, MJ Research) using the following protocol: 5' 95°C, 30 cycles of 30'' 95°C, 30'' 69.5°C, 30'' 72°C; stop at 4°C. The results were confirmed by sequencing.

Labelling. 100µl of each fluid (H_2O, motor-oil, paint) was labelled with 1µl 25fM of cloned logomer respectively. After 3 days logomers were re-extracted. Re-extraction from motor-oil (HD 10W-40, TS-Union) was done by adding 100µl of H_2O, thorough vortexing, 5' centrifugation at 20000g and collecting the aequous phase. Re-extraction from paint was done by adding 100µl turpentine, followed by thorough fragmentation and vortexing. After 5' centrifugation at 20000g the supernatant was collected. 100µl H_2O was added and vortexed thoroughly. After 5' centrifugation at 20000g the aequous phase was collected. In the case of paper-based materials 1µl of 25fM cloned logomer was dropped on surface of the paper and left for 3 days. For detection 1mm² of the labelled paper was cut out and used as template directly in a subsequent PCR.

Label detection. 1µl of each extraction volume, or 1mm² paper were used as template in a 50µl PCR reaction containing 35µl H2O, 5µl PCR Buffer 10x, 5µl $MgCl_2$ (25mM), 1µl dNTPs (10mM, Pharmacia Biotech), 0,5µl TAQ (5u/µl, Gibco-BRL), 1µl 5' start-primer (10µM), 1µl 3' end-primer (10µM) using the protocol: 5' 95°C, 30 cycles of 30'' 95°C, 30'' 65°C, 30'' 72°C; stop at 4°C.

Acknowledgements

We thank Libby Guethlein, Ulrich Boehm, Peter Dittrich and Elisabeth C. Proske for discussion and comments; Rita Lange and Sam Saghafi for technical assistance and the people in the lab in Cologne for their friendly support. The work was supported in parts by the Stifterverband der Deutschen Wissenschaft/Stiftung Winterling Marktleuthen.

Correspondence and requests for materials should be addressed to H.R.

(rauhe@LS11.cs.uni-dortmund.de)

References

1. Adleman, L. M. Molecular Computation of Solutions to Combinatorial Problems. *Science*, **266**, 1021-1024, (1994)
2. Winfree, E. Yang, X., Seeman, N. C. Universal Computation via Self-assembly of DNA: Some Theory and Experiments. *Proceedings of the 2nd DIMACS Meeting on DNA Based Computers, Princeton University, June 20-12*, (1996)
3. Winfree, E., Liu, F., Wenzler, L. A. & Seeman, N. C. Design and self-assembly of two-dimensional DNA crystals. *Nature*, **394**, 539-544, (1998)
4. Chomsky, N. On certain formal properties of grammars, *Inf. and Control.* **2:2**, 137-167, (1959)
5. Hopcroft, J. E., Ullman, J. D. Formal Languages And Their Relation To Automata. Addison-Wesley (1969)
6. Jeffreys, A. J. Minisatellite reapeat coding as a digital approach to DNA typing. *Nature*, **354**, 204-209, (1991)
7. Feldkamp, U., Ein DNA-Sequenz-Compiler, Diploma Thesis, Dept. of Computer Science LS11, University of Dortmund, Germany (1999)
8. Capecchi M.R. Altering the genome by homologous recombination. *Science*, **244**(4910), 1288-1292, (1989)
9. Clelland C. T., Risca, V., Bancroft, C. Hiding messages in DNA microdots. *Nature*, **399**, 533-534, (1999)
10. Richter, C., Leier, A., Molekulare Kryptographiesysteme. Diploma thesis at the department of computer science, University of Dortmund.
11. Leier, A., Richter, C., Banzhaf, W., Rauhe, H. DNA steganography with DNA binary strands. (submitted)
12. Press, W. H., Teukolsky, S. A., Vetterling, W. T., Flannery, B. P. Numerical Recipes in C, 2nd Edition, 274-328, Cambridge University Press (1992)
13. The Quantum World Corporation, http://www.comscire.com/QNGINFO.html (1995)
14. Chen, J., Seeman, N.C. Synthesis from DNA of a molecule with the connectivity of a cube. *Nature*, **350**, 631-633, (1991)
15. Braun, E., Eichen Y., Sivan, U., Ben-Yoseph, G. DNA-templated assembly and electrode attachment of an conducting silver wire. *Nature*, **391**, 775-778, (1998)
16. Fink, H.-W., Schönenberger, C. Electrical conduction through DNA molecules. *Nature*, **398**, 407-410, (1999)
17. Burke DT, Carle GF, Olson MV, Cloning of large segments of exogenous DNA into yeast by means of artificial chromosome vectors. *Science* , **236**(4803):806-12, (1987)
18. Gardner, T.S., Cantor, C. R., Collins, J. J. Construction of a genetic toggle switch in Escherichia coli. Nature, **403**(6767), (2000)