

A Self-Repairing and Self-Healing Electronic Watch: The BioWatch

André Stauffer, Daniel Mange, Gianluca Tempesti, and Christof Teuscher

Logic Systems Laboratory, Swiss Federal Institute of Technology,
CH-1015 Lausanne, Switzerland,
`andre.stauffer@epfl.ch`

Abstract. The Embryonics project is inspired by some of the basic processes of molecular biology, such as the embryonic development of living beings. Transposing these processes in to digital electronic integrated circuits, we design artificial organisms endowed with properties typical of to the living world, such as self-repair and self-healing. In order to illustrate the original features of the Embryonics project, we define the cellular and molecular architecture of a giant artificial organism, the BioWatch. The hardware implementation of a microprogrammed version of our watch exploits a new reconfigurable tissue, the bio-inspired electronic wall or BioWall.

1 Introduction

1.1 Embryonics = embryonic electronics

The *Embryonics* project (for *embryonic electronics*) is inspired by the basic processes of molecular biology and by the embryonic development of living beings [13], [3]. By adopting certain features of cellular organization, and by transposing them to the two-dimensional world of integrated circuits on silicon, we have already shown that properties unique to the living world, such as *self-replication* and *self-repair*, can also be applied to artificial objects (integrated circuits) [4].

Our final objective is the development of very large scale integrated (VLSI) circuits capable of self-repair and self-replication. Self-repair allows partial reconstruction in case of a minor fault, while self-replication allows the complete reconstruction of the original device in case of a major fault [4].

1.2 A bio-inspired watch: the BioWatch

In order to illustrate to the general public the original features of the Embryonics project, we decided to realize a giant electronic watch capable of self-repair and self-healing. In addition to its pedagogical virtues, this project introduces a new concept, the *reconfigurable computing tissue*, that binds tightly together reprogrammable logic circuits (FPGAs), input units (touch-sensitive buttons) and output units (LED-matrix displays).

Conception, birth, growth, maturity, illness, old age, death: this is the life cycle of living beings. The proposed demonstration will stage the life cycle of the BioWatch from conception to death. Visitors will face a large wall made up of a mosaic of many thousands of transparent electronic modules or *molecules*, each containing a display. At rest, all the modules will be dark. A complex set of signals will then start to propagate through the space (conception) and program the modules to realize the construction of a beating electronic watch (growth). Visitors will then be invited to attempt to disable the watch: on each molecule, a push-button will allow the insertion of a fault within the module (wounding) or its removal from the module (healing). The watch will automatically repair after each aggression (cicatrization). When the number of faults exceeds a critical value, the watch dies, the wall plunges once more into darkness, and the complete life cycle begins anew.

The first part of this paper is devoted to a macroscopic description of the Biowatch project. Section 2 describes the two architectural features of our artificial organism: multicellular organization (the organism consists of an array of identical physical elements, the cells) and cellular differentiation (each cell contains the complete blueprint of the organism, that is, its genome, and specializes depending on its position within the array). While maintaining the correct time, our multicellular organism is capable of self-repair (it can automatically replace one or more faulty cells) and self-healing (it can automatically recover one or more cells, should the fault disappear).

The microscopic structure of the cell relies on three fundamental features: multimolecular organization (the cell is itself decomposed into an array of physically identical elements, the molecules), molecular configuration (the boundaries between cells and the position of spare molecules are defined thanks to a cellular automaton), and fault detection within each molecule leading to the self-repair of the cell (through the replacement of the faulty molecules). This structure is described in the second part of this paper (Section 3). The current implementation of our reconfigurable computing tissue, the *BioWall* (*bio-inspired electronic wall*), forms the core of Section 4, which constitutes in fact a description of the molecular level of our system.

This paper presents a firmware implementation of the bio-inspired electronic watch based on an array of small processors. A fully hardware implementation of this watch is reported elsewhere [10].

2 BioWatch's macroscopic description: an artificial organism

In the framework of electronics, the environment in which our quasi-biological development occurs consists of a finite (but as large as desired) two-dimensional space of silicon. This space is divided into squares or *cells*. Since such cells (small processors and their memory) have an identical physical structure, i.e. an identical set of logic operators and of connections, the cellular array is homogeneous.

As the program in each cell (our artificial genome) is identical, only the state of the cell, i.e. the contents of its registers, can differentiate it from its neighbors.

Our artificial organism is designed to count and display hours, minutes, and seconds, from 00h00'00" to 23h59'59". The input signal used for synchronizing the units of seconds is delivered by wireless broadcast.

2.1 Multicellular organization

Multicellular organization divides the artificial organism (*ORG*) into a finite number of cells (Figure 1), where each cell (*CELL*) realizes a unique function, defined by a sub-program called the *gene* of the cell. The same organism can contain multiple cells of the same kind (in the same way as a living being can contain a large number of cells with the same function: nervous cells, skin cells, liver cells, etc.). Moreover, each cell is associated with some *output state*.

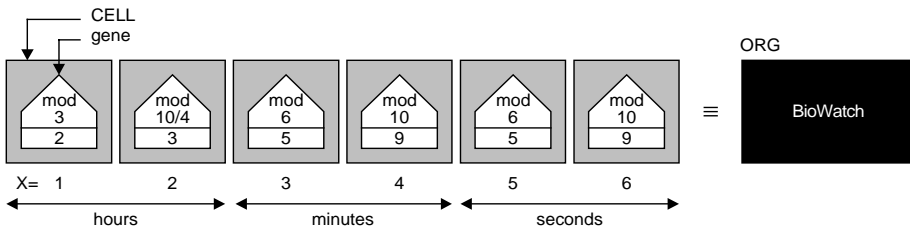


Fig. 1. Multicellular organization of the BioWatch organism. In this example, BioWatch displays 23h59'59".

Our *BioWatch* is thus a one-dimensional artificial organism implemented with six cells and featuring four distinct genes (“mod 10” for counting the units of seconds or minutes, “mod 6” for counting the tens of seconds or minutes, “mod 10/4” for counting the units of hours depending on the value of the tens of hours, and “mod 3” for counting the tens of hours). The output state is the current value of the elapsed time and varies from 0 to 9 (for units of seconds, minutes, and hours), from 0 to 5 (for tens of seconds and minutes), and from 0 to 2 (for tens of hours).

2.2 Cellular differentiation

Let us call *operative genome (OG)* a program containing all the genes of an artificial organism, where each gene is a sub-program characterized by a set of instructions and by the position of the cell (its coordinates X, Y). Figure 1 then shows the operative genome of BioWatch, with the corresponding horizontal (X) coordinate; the vertical (Y) coordinate can be ignored in this particular unidimensional case. Let then each cell contain the entire operative genome *OG*

(Figure 2a): depending on its position in the array, i.e. its place in the organism, each cell can interpret the operative genome and extract and execute the gene which defines its function.

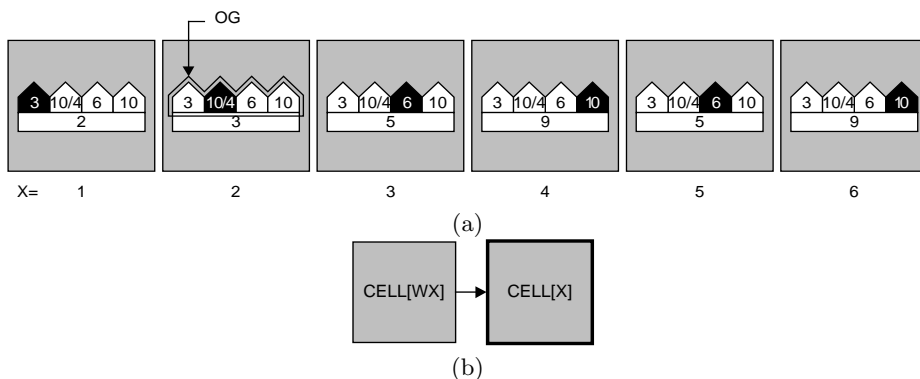


Fig. 2. Cellular differentiation of the BioWatch organism. (a) Global organization; *OG*: operative genome (genes and coordinates). (b) Central cell $CELL[X]$ with its west neighbor $CELL[WX]$.

In summary, storing the whole operative genome in each cell makes the cell universal: given the proper coordinates, it can execute any one of the genes of the operative genome and thus implement *cellular differentiation*. In our artificial BioWatch, any cell $CELL[X]$ computes its coordinate X by incrementing the coordinate WX of its neighbor immediately to the west (Figure 2b). Any cell $CELL[X]$ can thus be formally defined by a set of modulo-counting sub-programs (its operative genome *OG*) and by its coordinate X . In the case of BioWatch, we have the operative genome of Figure 3.

```

OG: operative genome
X = WX+1
case of X:
  X = 1: count_mod_3      (tens of hours)
  X = 2: count_mod_10/4  (units of hours)
  X = 3: count_mod_6     (tens of minutes)
  X = 4: count_mod_10    (units of minutes)
  X = 5: count_mod_6     (tens of seconds)
  X = 6: count_mod_10    (units of seconds)
end case
    
```

Fig. 3. The operative genome *OG* of the BioWatch organism (first part).

2.3 Self-repair and self-healing properties

In order to implement *self-repair* and *self-healing*, we need to add *spare cells* (*SC*) to the right of the original unidimensional organism (Figure 4). These cells are defined by the coordinate $X = 7$.

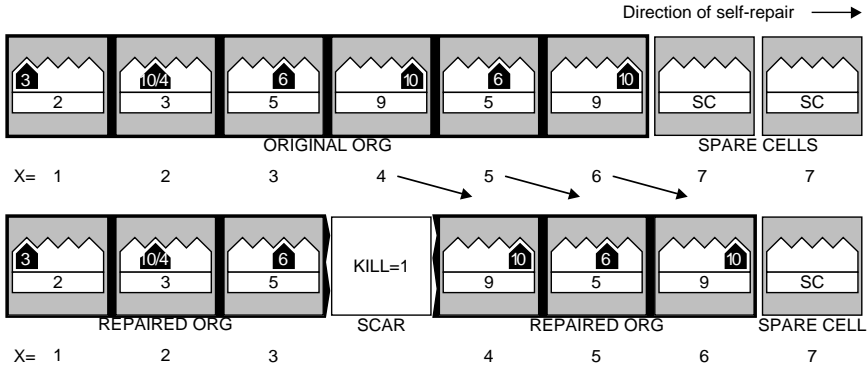


Fig. 4. Self-repair of the 6-cell BioWatch organism with two spare cells and one faulty cell; SC = spare cell.

The existence of a fault is detected by a *KILL* signal which is produced at the cellular level (see Section 3). The state $KILL = 1$ identifies the faulty cell which is deactivated (column $X = 4$ in Figure 4). All the functions (X coordinate and gene) of the cells at the right of the column $X = 3$ are shifted by one column to the right. Obviously, this process requires as many spare cells, to the right of the array, as there are faulty cells to repair (two spare cells tolerating two successive faulty cells in the unidimensional example of Figure 4). It also implies that the organism must have the capability of bypassing the faulty cell and to divert to the right all the required signals (such as the operative genome and the X coordinate, as well as the data busses).

The disappearance of the *KILL* signal ($KILL = 0$) means that the faulty cell (column $X = 4$ in Figure 5) has recovered all its functionalities at the cellular level and can be reactivated. All the coordinates and the genes of the cells at its right are then shifted by one column to the left and our organism possesses two spare cells again. While performing the self-repair and self-healing processes, the BioWatch maintains the correct time (23h59'59" in Figures 4 and 5).

3 BioWatch’s microscopic description: the artificial cells

In each cell of every living being, the genome is translated sequentially by a chemical processor, the *ribosome*, to create the proteins needed for the organism’s survival. The ribosome itself consists of molecules, whose description is an important part of the genome.

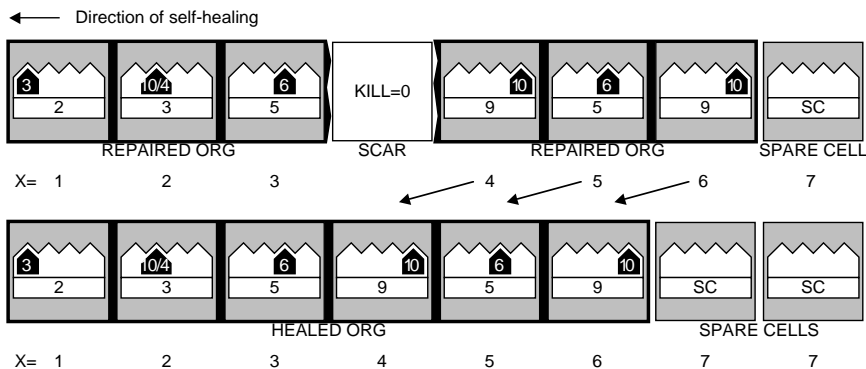


Fig. 5. Self-healing of the 6-cell BioWatch organism after a faulty cell is recovered.

As mentioned previously, in the Embryonics project each cell is a small processor or *binary decision machine*, sequentially executing the instructions of our artificial genome, the operative genome *OG* (Figure 3). The need to realize organisms of varying degrees of complexity has led us to design an artificial cell characterized by a *flexible architecture*, that is, itself configurable. It will therefore be implemented using a new kind of fine-grained field-programmable gate array (FPGA). We will illustrate the use of this FPGA through the design of the current cell of our BioWatch, a *shift binary decision machine*.

3.1 Multimolecular organization

Each element of our FPGA is then equivalent to a *molecule*, and an appropriate number of these artificial molecules allows us to realize application-specific processors. We will call *multimolecular organization* the use of many molecules to realize one cell.

A consequence of our choices is that we require a methodology to generate, starting from a set of specifications, the configuration of our FPGA, consisting of a homogeneous network of molecules, defined by an identical architecture and a usually distinct state (the *molecular code*, or *MOLCODE*).

To fulfill this requirement, we have selected a particular representation: the *ordered binary decision diagram* (OBDD) [1], [2], [7]. This representation, with its well-known intrinsic properties such as canonicity, was chosen for two main reasons.

1. It is a graphical representation which exploits well the two-dimensional space and immediately suggests a physical realization on silicon.
2. Its structure leads to a natural decomposition into molecules realizing a logic test, easily implemented by a multiplexer.

The reconfigurable molecule, henceforth referred to as MUXTREE (for *multiplexer tree*), consists essentially of a programmable multiplexer (with one control

variable), a D-type flip-flop, and a switch block allowing all possible connections between two horizontal and two vertical long-distance busses. The behavior of a MUXTREE molecule, described in detail elsewhere [6](pp. 135-143), [4], is completely defined by a molecular code organized as a 20-bit data $MOLCODE_{19:0}$, itself stored in a *configuration register* CREG. Depending on the value of a 1-bit *operation code* OPC, the MUXTREE molecule can be used in two basically different modes.

1. The *logical* mode ($OPC = 0$), in which all the logic functions (multiplexer, D-type flip-flop) and all the connections are activated.
2. The *memory* mode ($OPC = 1$), in which part of the configuration register CREG (the eight bits $MOLCODE_{7:0}$ or the sixteen bits $MOLCODE_{15:0}$) becomes a shift register that will be used in the implementation of the shift memory of the processor.

Our artificial cell embeds a special kind of binary decision machine, a *shift binary decision machine* [6](pp. 259-265), with a shift memory capable of storing 128 12-bit micro-instructions for the operative genome OG . Assembling eight such cells allows us to realize the current version of BioWatch (Figure 4). The specifications of the organism are described in the program of Figure 3. The operative genome OG consists of four distinct genes whose execution depends solely on the X coordinate. A fifth gene is needed only to identify the cells which are *spare cells* (SC).

The shift binary decision machine is specially designed to fit into an array of MUXTREE molecules: due to the difficulty of embedding a classic random access memory (RAM) in such an array (mainly due to the excessive number of molecules needed for decoding the RAM address), the actual program memory, or *shift memory*, consists of shift registers implemented using the MUXTREE molecules in memory mode.

Figure 6 shows the final BioWatch cell, connecting the shift binary decision machine SBDM and the shift memory SMEM. This structure involves the following hardware resources:

- a 7-bit current time register $REG(Q1 : 7)$;
- a 7-bit duplicate time register $REG(D1 : 7)$;
- a 3-bit coordinate register $REG(X2 : 0)$;
- a 1-bit test register $REG(T)$;
- a 1-bit healing register $REG(H)$;
- a 1-bit ripple count circuit $RCC(RC)$.

The 7-bit coding of the time data is chosen in order to insure full compatibility with a 7-segment display. The operative genome OG also includes a control part that regulates the operations executed by the registers (Figure 7). As shown in the figure, these operations depend on the values of the test and healing bits of the cell itself (T , H), of the first neighboring cell to the west (WT , WH), and of the first neighboring cell to the east (EH). In the normal time counting mode ($WH = 0$, $H = 0$, $WT = T'$), the duplication of the current time is effected

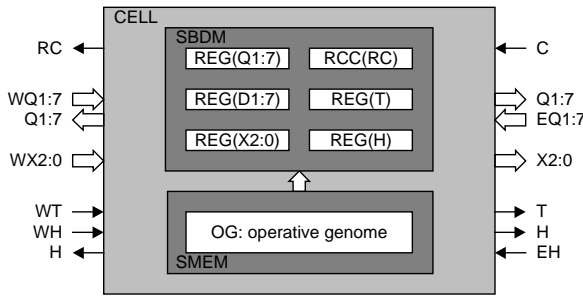


Fig. 6. Block diagram of the BioWatch artificial cell; SBDM: shift binary decision machine; SMEM: shift memory.

at each program execution cycle (Figure 8). When a faulty cell is detected by its right-hand neighbor ($WH = 0, H = 0, WT = T$), a multicycle self-repair process (Figure 9) sequentially copies the duplicate time and thus allows our BioWatch to maintain the current time. When the faulty cell recovers ($H = 1$), a multicycle self-healing process (Figure 10) shifts the current time sequentially back to the left.

```

OG: operative genome
case of WH,H,EH:
  WH,H,EH = 00-:
    case of WT,T:
      WT,T = 00:  Q = D
                  T = 1
      WT,T = 01:  D = WQ
      WT,T = 10:  D = WQ
      WT,T = 11:  Q = D
                  T = 0
    end case
  WH,H,EH = 011: D = EQ
                  Q = D
                  H = 0
  WH,H,EH = 1--: H = 1
end case
    
```

Fig. 7. The operative genome *OG* of the BioWatch organism (second part).

Based on the OBDD decomposition of its resources, the MUXTREE implementation of the BioWatch artificial cells leads to the 24×25 array of molecules shown in Figure 11. As can be seen in the center of the figure, the 7-segment display of the current time data is part of the array. This is done in order to take full advantage of the reconfigurable computing tissue introduced in Section 4. The seven lower rows of the MUXTREE implementation constitute the 128×12 -

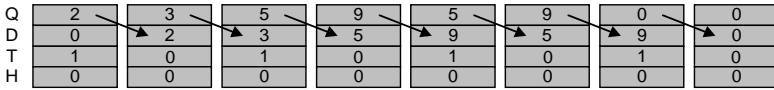


Fig. 8. Duplication of the BioWatch’s current time (when $T \neq WT$ then D becomes WQ).

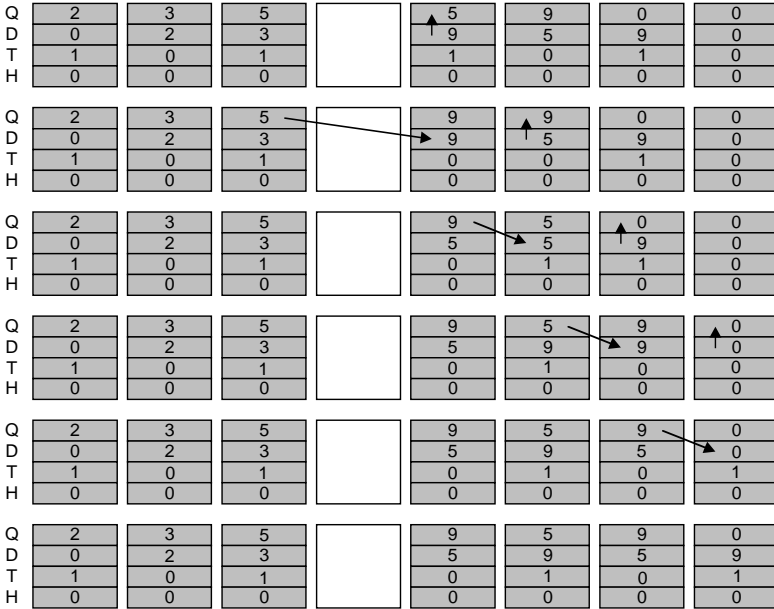


Fig. 9. Time-preserving self-repair process of the BioWatch organism (when $T = WT$ then Q becomes D).

bit shift memory SMEM needed to store the 126 12-bit word instructions of the complete operative genome (Figures 3 and 7).

To sum up, *multimolecular organization* divides finally the BioWatch artificial cell, our shift memory binary decision machine, into a finite number of molecules (600), where each molecule is defined by a unique configuration, its molecular code $MOLCODE_{19} : 0$. The *ribosomic genome* (RG) is then the sum of the 600 $MOLCODEs$ of the 600 active MUXTREE molecules.

3.2 Molecular configuration

The information contained in the $MOLCODE$ defines the logic function or the memory contents of each molecule. To obtain a functional cell, i.e. an assembly of MUXTREE molecules, we require two additional pieces of information, defining

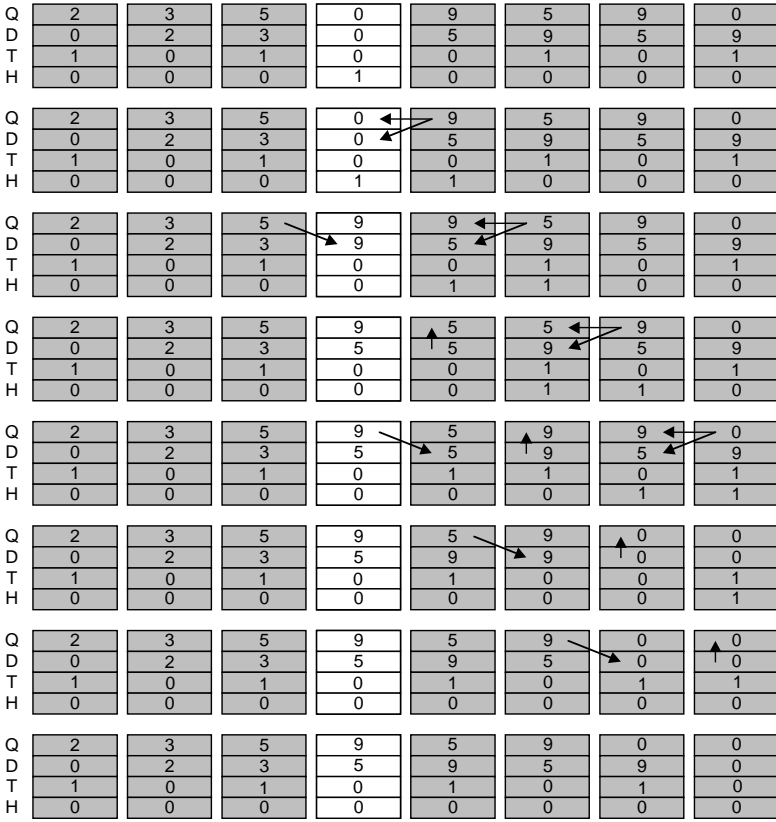


Fig. 10. Time-preserving self-healing process of the BioWatch organism (when $WH, H, EH = 011$ then D and Q become EQ).

the physical position of each molecule within a cell and the presence and position of the spare columns required by the self-repair mechanism (Subsection 3.3).

The mechanism which we have adopted consists of introducing in the FPGA a regular network of automata (state machines) called *space divider* [6], [11], [12]. We call *polymerase genome* PG the sequence of states that needs to be applied to the lower left hand automaton in order to divide the entire space into cells. If the states C , V , and H represent respectively corner, vertical, and horizontal boundaries, the sequence:

$$PG = C, V * [h - 1], H * [w - 1], C, \dots \tag{1}$$

defines a cell h molecules high and w molecules wide. In this relation, where the notation $X * [n]$ represents the state X repeated n times, the presence of spare columns is indicated by replacing one or more occurrences of H by S . The

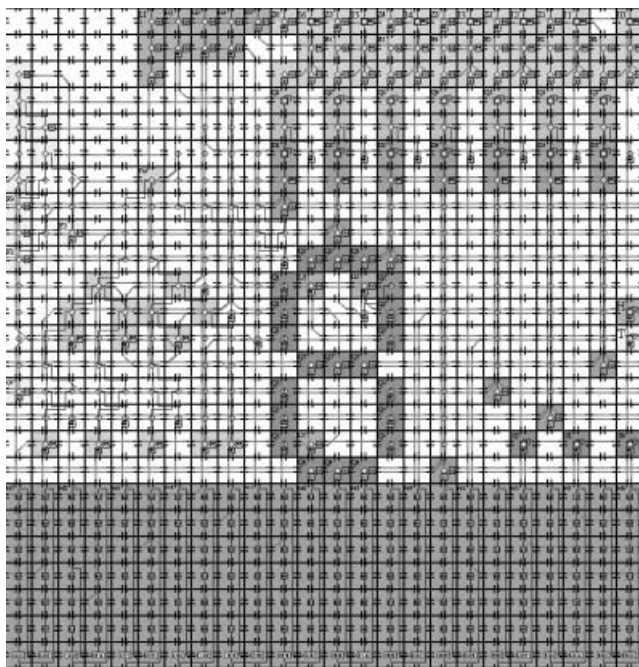


Fig. 11. The $24 \times 25 = 600$ MUXTREE molecule array implementing the BioWatch's artificial cell; the seven lower rows of molecules constitute the 128×12 -bit shift memory.

BioWatch artificial cell shown under construction in Figure 12 with two columns of spare molecules results therefore from a cycle of the following states:

$$PG = C, V * 24, H * 7, S, H * 8, S, H * 8, S, C, \dots \quad (2)$$

The details of the design of the space divider are described elsewhere [4].

3.3 Molecular fault detection

The specification of the molecular self-repair system must include the following features:

- it must operate in real time, in response to the activation of the push-button included in each molecule;
- it must preserve the memorized values, that is, the state of the D-type flip-flop and the *MOLCODE* of each molecule;
- it must assure the automatic repair at the cellular level;
- in case of multiple faults, it must generate a global signal $KILL = 1$ that activates the suppression of the cell and starts the self-repair process of the

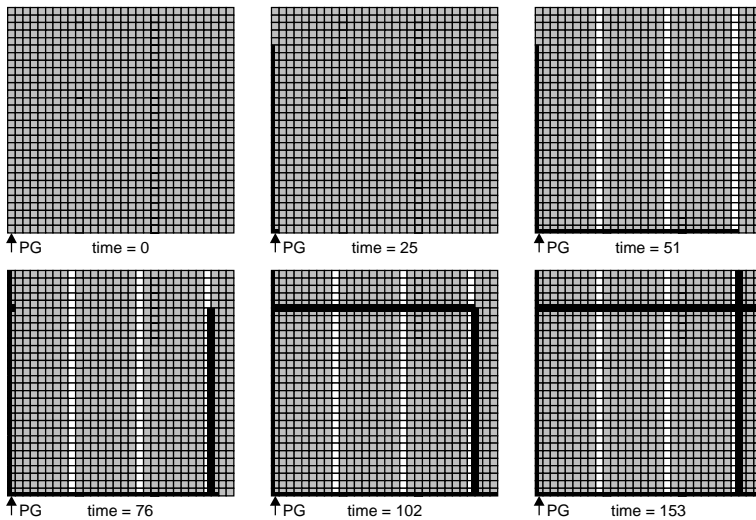


Fig. 12. The BioWatch's space divider (height=25, width=27, 1 spare column out of 9).

complete organism, or $KILL = 0$ which activates the self-healing process (Subsection 2.3).

To meet the specifications, and in particular the requirement that the hardware overhead be minimized, our self-repair system exploits the programmable frequency and distribution of the spare columns (Subsection 3.2) by limiting the reconfiguration of the array to a single molecule per line between two spare columns (Figure 13). This choice allows us to minimize the amount of logic required for the reconfiguration of the array, while keeping a more than acceptable level of robustness.

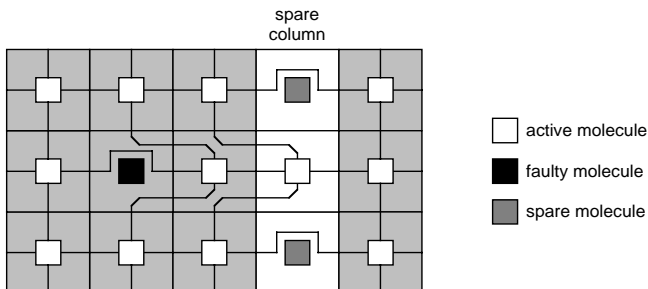


Fig. 13. The self-repair mechanism for an array of MUXTREE molecules.

4 Conclusion: toward a bio-inspired electronic wall (BioWall)

The *BioWall* (bio-inspired electronic wall) [5] is an ongoing project in our laboratory. This wall is intended to be a reconfigurable computing tissue capable of interacting with its environment by means of a large number of touch-sensitive elements coupled with two-color LED displays. Figure 14 shows a prototype of its hardware implementation. The $200 \times 25 = 5000$ molecules of its final implementation will allow us to configure the six cells of our BioWatch organism and two extra spare cells. Each molecule is made up of a transparent touch-sensitive element, a two-color 8×8 dot-matrix LED display, and a reconfigurable Xilinx Spartan XCS10XL FPGA circuit (Figure 15). Within the molecule, the transparent touch-sensitive element and the LED display are physically joined by an adhesive film. As each of the molecules provides the same connections to its four direct neighbors, the BioWall is homogeneous and fully scalable.

In the BioWatch application (Figure 16), the touch-sensitive element of the BioWall molecule acts as a push-button used to render the molecule faulty or healthy again. The LED display shows the boundaries, the spare columns and the current time of the cell, as well as the faulty or healthy state of the molecule. The configuration of the Xilinx FPGA implements all the MUXTREE molecule specifications in the BioWall molecule.

The programmable robustness of our system depends on a redundancy (spare molecules and cells) that is itself programmable. This feature is one of the main original contributions of the Embryonics project. It makes possible to program (or reprogram) a greater number of spare molecules and spare cells for operation in hostile environments (e.g., space exploration). A detailed mathematical analysis of the reliability of our systems has been carried out at the University of York [8], [9].

With respect to this design process, the programming of the molecular array of MUXTREE elements, our reconfigurable tissue, takes place in the following order.

1. The polymerase genome (*PG*) is injected in order to set the boundaries between cells and define the spare columns.
2. The ribosomic genome (*RG*), i.e. the string of $24 \times 25 = 600$ *MOLCODES*, is injected in order to configure the BioWatch cell and to fix the final architecture of the shift binary decision machine inside the cell.
3. The operative genome (*OG*) is stored within the shift memory of each cell in order for it to execute the counting, repairing, and healing specifications.

Echoing biology, we have faced complexity by decomposing the organism into cells and then the cells into molecules. This decomposition implies three configuration steps: the polymerase genome organizes the space by defining the cells' boundaries, the ribosomic genome defines the architecture of each cell as an array of molecules, and finally the operative genome makes up the program that will be executed by the cellular processors to accomplish the required task. The Latin motto "divide and conquer" maintains its relevance even today.

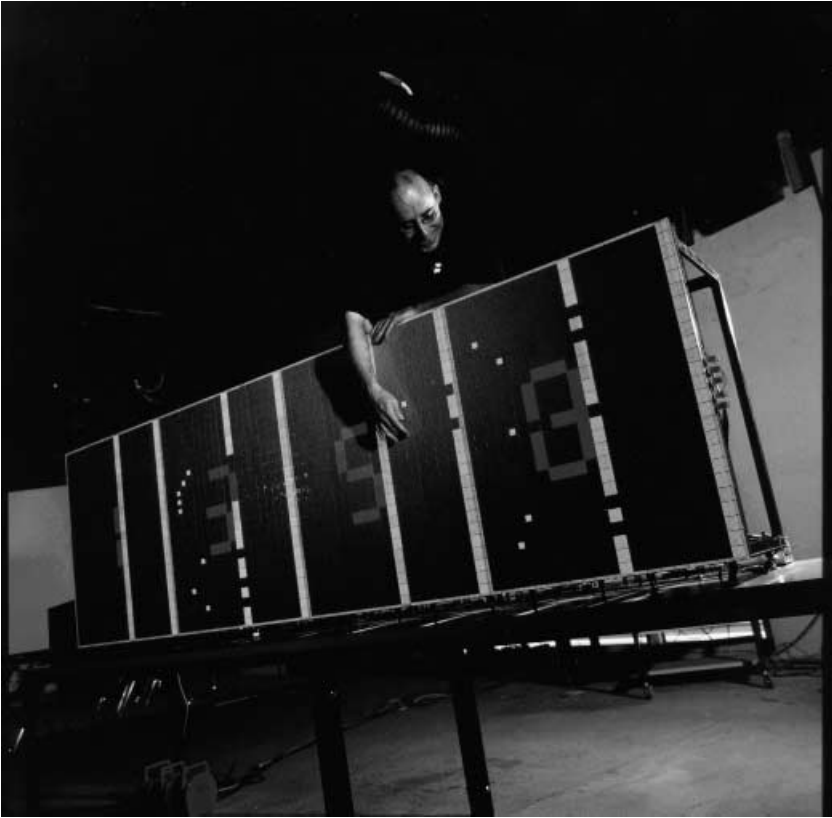


Fig. 14. The present BioWall prototype consisting of about $80 \times 25 = 2000$ molecules (Photograph by A. Herzog).

Acknowledgments

This work was supported in part by the Swiss National Foundation under grant 21-54113.98, by the Leenaards Foundation, Lausanne, Switzerland, and by the Villa Reuge, Ste-Croix, Switzerland.

References

1. S. B. Akers. Binary decision diagrams. *IEEE Transactions on Computers*, c-27(6):509–516, June 1978.
2. R. E. Bryant. Symbolic boolean manipulation with ordered binary-decision diagrams. *ACM Computing Surveys*, 24(3):293–318, 1992.
3. D. Mange, M. Sipper, and P. Marchal. Embryonic electronics. *BioSystems*, 51(3):145–152, 1999.

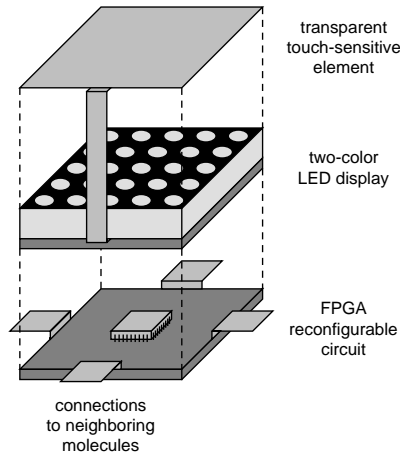


Fig. 15. The basic molecule of the reconfigurable computing tissue.



Fig. 16. The BioWall implementation of the 8-cell BioWatch (Computer graphic by E. Petraglio).

4. D. Mange, M. Sipper, A. Stauffer, and G. Tempesti. Toward robust integrated circuits: The Embryonics approach. *Proceedings of the IEEE*, 88(4):516–541, April 2000.
5. D. Mange, A. Stauffer, G. Tempesti, and C. Teuscher. Tissu électronique reconfigurable, homogène, modulaire, infiniment extensible, à affichage électro-optique et organes d’entrée, commandé par des dispositifs logiques reprogrammables dis-

- tribués. Patent pending, 2001.
6. D. Mange and M. Tomassini, editors. *Bio-Inspired Computing Machines*. Presses polytechniques et universitaires romandes, Lausanne, 1998.
 7. C. Meinel and T. Theobald. *Algorithms and Data Structures in VLSI*. Springer-Verlag, Berlin, Germany, 1998.
 8. C. Ortega and A. Tyrrell. Reliability analysis in self-repairing embryonic systems. In A. Stoica, D. Keymeulen, and J. Lohn, editors, *Proceedings of The First NASA/DOD Workshop on Evolvable Hardware*, pages 120–128, Pasadena, CA, 1999. IEEE Computer Society.
 9. C. Ortega and A. Tyrrell. Self-repairing multicellular hardware: A reliability analysis. In D. Floreano, J.-D. Nicoud, and F. Mondada, editors, *Proceedings of the 5th European Conference on Artificial Life (ECAL'99)*, Advances in Artificial Life. Springer-Verlag, Berlin, 1999.
 10. A. Stauffer, D. Mange, G. Tempesti, and C. Teuscher. Biowatch: A giant electronic bio-inspired watch. In A. Stoica, D. Keymeulen, J. Lohn, and R. Zebulum, editors, *Proceedings of The Third NASA/DOD Workshop on Evolvable Hardware*, Long Beach, CA, 2001. IEEE Computer Society.
 11. G. Tempesti. *A Self-Repairing Multiplexer-Based FPGA Inspired by Biological Processes*. PhD thesis, Computer Science Department, Swiss Federal Institute of Technology Lausanne, 1998.
 12. G. Tempesti, D. Mange, and A. Stauffer. Self-replicating and self-repairing multicellular automata. *Artificial Life*, 4(3):259–282, 1998.
 13. L. Wolpert. *The Triumph of the Embryo*. Oxford University Press, New York, 1991.