Despite ground breaking advances in most engineering and scientific disciplines during the past decades, reliability engineering has not seen significant breakthroughs or noticeable advances. In order to guarantee failure free system operation engineers still apply the same established principles that would ensure continued proper operation in the presence of a fault. This would either be graceful degradation where quality of system operation proportionally decreases with the severity of the failure or incorporating redundancies. Triple modular redundancy [TMR] is still one of the most prevalent techniques used today on the system level where three systems are replicated that in the presence of a single fault would still guarantee failure free operation. But very high cost is associated with replicating entire systems.

Nearly 90% of the system crashes are attributed to 'soft' transient faults where only the memory content of the system mutates that in turn causes system to malfunction. These errors are induced by temporary environmental conditions, such as cosmic rays and electromagnetic interference. Such errors are most likely to happen in critical applications where we cannot afford for an error to occur. Engineers tried many methods like the TMR and DMR [Double modular redundancy] but it increased the complexity and cost of the systems. And when it comes to innovation, invention and transformation, nature can be excellent place to start. Historically a source of inspiration for many disciplines (witness man's early attempts to fly using contraptions that imitate birds); in recent times nature has become a growing driver in the design and creation of electronic devices. VLSI circuits have been extensively used in almost all applications because of its reduced size, cost, reliability and its very low power consumption. Many bio-inspired changes have been made in VLSI circuits to make it fault tolerant and among them the simplest one which is being inspired from unicellular electronics has been chosen as the 'man of act' here.

II. METHODS AND MATERIAL

A. Errors in VLSI Circuits

Failures in VLSI systems could result from varied types of faults that can be classified as either soft (transient) or permanent (hardware) ones. Transient faults are induced by temporary environmental conditions, such as cosmic rays and electromagnetic interference and could for example cause information mutation in memory elements. Permanent faults are the result of irreversible device and circuit changes, such as the following:

a. Electro migration, which causes thinning and eventual open circuit of metal tracks.
b. Hot carrier effect, which causes shift in device threshold voltage and its conveyed conductance.
c. Time dependent dielectric breakdown, which causes gate oxide to substrate short – circuit.

B. Bio-Inspired Fault Tolerance

Nature offers to us some remarkable examples of how to deal with complexity and it associated unreliability. For example, human body is one of the most complex systems ever known. Local failures are common, but the overall function of our organism is so reliable because of the self-diagnosis and self-healing mechanisms that work ceaselessly throughout our bodies. These mechanisms are the result of millions of years of our genes' evolution.

During the past few years the work did on bio-inspired systems have generated some remarkable results. Genetic algorithms, neural networks, artificial brains and evolvable hardware and only few of them. What attracts scientists and engineers to nature lies in the characteristics biological organisms' possess. These characteristics include evolvability, multi cellular structures, auto regulation and learning that allow them to adapt to the changes in their living environment.

A recent approach to fault tolerance is borrowing from nature the main principles that make living things so resilient to faults. Mechanisms such as self-diagnosis, self-healing, reproduction and adaptation are being transported to the arena of electronics. All these characteristics seem to be a natural consequence of the massively parallel arrays of cells that constitute every living being. The following section discusses the various bio-inspired fault tolerant techniques used in VLSI circuits.

C. Prokaryotic Biological Systems

The structure and behavior of prokaryotic cells have a number of features which we have utilized within the work presented here. Prokaryotes are simple unicellular forms of life. Prokaryotic cells are much simpler than eukaryotic cells, requiring far fewer genes in their genome in order to function. Whereas eukaryotic DNA contains coding genes for all possible cell functions within a multicellular organism the genes in prokaryotic DNA only need to code for proteins needed by that particular cell. Their DNA is still a duplexed structure, with two strands where one a complemented version of the other. Bonds form between paired genes in the strands, resulting in the DNA forming a double helix. Damage to one of the strands of DNA can be detected by enzymes and the undamaged strand is then used to repair the damaged one.

Another important characteristic of prokaryotes is their ability to learn from and save their environmental experience and then transfer such change of their genetic material to other cells via a process called horizontal gene transfer (HGT). This simple process inherently adds valuable features to the natural creature such as adaptability, evolvability, and resistance against environmental attacks and allows newly recruited cells to be endowed with required genetic properties.

D. Unitronics: A New Bio-Inspired Artificial Organism

The prokaryotic bio-inspired model refers to a community of prokaryote inspired electronic cells; hence, Unitronics from Unicellular electronics. Unitronics is an array of digital electronic cells with mechanisms inspired by unicellular life. The artificial bio-inspired prokaryotic system proposes a multilayered hierarchical architecture similar to those found in biological prokaryotic systems. This is shown in Table 1 below.

A Unitronics (UX) cell array is made up of two different types of cells; core cells comprising the main body of the array, which are surrounded by peripheral cells around the perimeter as shown in Fig. 1. Core cells are configured to implement specific data routing and processing functions, as defined by the genes stored in their configuration registers; additionally they are used to configure the internal data bus. Peripheral cells manage the data flow to and from outside the array via a peripheral bus.

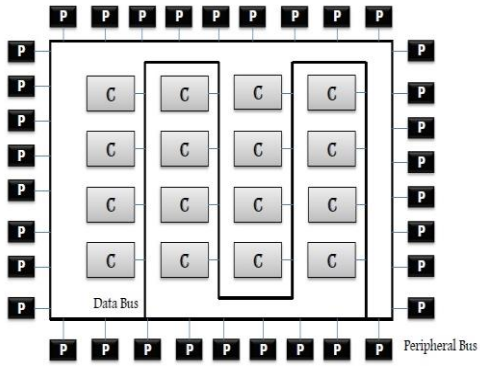

Figure 1: Schematic of a Unitronics cell array, Core cells are denoted C, and peripheral cells P.

i. Core Cell Architecture

Core Cells [C-Cells] are the processing and communication elements of the system. The core cell comprises:

1) A memory, which in turn is built up of two registers:
   a) a configuration register that stores the cell's operational gene;
   b) a non-configuration register that facilitates data back-up and self-repair;
2) A universal function unit (FU) with two programmable bit-slice blocks;
3) A communication unit that incorporates a connection box (CB) and a switch block (SB) to select and route signals.

ii. Peripheral Cell

Peripheral cells are much simpler than core cells; their role is simply to provide a configurable connection to external I/O channels via a peripheral bus. They also fulfill a vital role within the built-in self-test mechanism used in the UX array.

E. Self-Repair Mechanism

The self-repair process consists of two parts: topological and process-related. The former refers to the structure of the hardware and specifically to the bus layer that facilitates repositioning of cells if a faulty cell is detected. The latter is associated with the compression/correlation mechanism and colonial growth morphology, which is a process that calculates the recoverable gene from redundant ones. The fault repair process is controlled by a very small external controller that monitors all the cells in the community and makes spare cells available when needed.

The fault repair procedure is as follows.

a. Shift: A sweeping process will check a flag in each cell of the Unitronic array to determine whether it indicates a fault. If one is detected, a spare healthy cell at the end of the chain is released. It will then acquire the gene from its closest neighboring cell, which in turn will take the gene from its adjacent cell. This process ripples back in the chain of cells and will eventually vacate a "stem" cell just adjacent to the faulty one as shown in Fig. 2. The sweeping process repeats until all faulty cells are replaced. The advantage of such shifting is that regardless of the location of faults, provided there are sufficient spare cells, repair is always possible. This is in contrast to cold redundancy schemes where for example in the case of row/column shifting sufficient spares must be available in the faulty cells' row/column.
b. A process similar to that for soft faults is carried out for each of the damaged cells, with their reconstructed gene being loaded into the new cell.

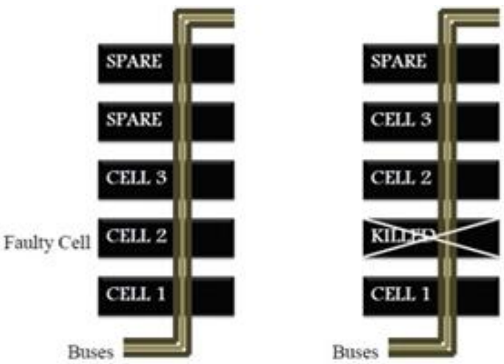

Figure 2: Elimination of the faulty cell

F. Built-In Self-Test

The bio-inspired self-test we are proposing is based on two characteristics of biological systems.

a. In nature, the DNA is a double helix, a duplicated sequence of complementary genes. It means that both sequences define exactly the same organism with exactly the same features. Therefore one strand is sufficient for the growth and development of an organism.
b. Transposons (formally termed jumping genes) are sequences of DNA that can move around to different positions within the genome of a cell. Such mobile genetic elements can be grouped and can move within the genome from one position to another using a "cut and paste" mechanism.

These two characteristics found in nature can be inspirational in the development of bio-inspired self-tests for artificial systems by observing the following:

a) If we could guarantee that by configuring the processing elements of the cell with its gene and complementary gene, their functionality would remain the same;

b) That the concept of the jumping genes mechanism could offer a solution to switch over and substitute input signals of such processing elements and interchange their outputs.

A simplified FU of a core cell is shown in Fig. 3. Multiplexers 1, 2, and 3 are used for input signal routing for the 2:1 functional multiplexer (Mux 5). This can configure the cell to implement any Boolean function of 2-input variables. Mux 4 is a mode selector, that provides selection between test (T = "0") and normal (T = "1") modes of operation. This simplified core cell satisfies both biological characteristics to provide sufficient self-test. The genome sequence is S5, S4, S3, S2, S1, and S0. In test mode, like a double helix, all bits are complemented. Also S3S2 and S1S0 are "jumped" so that their respective places in the cell's gene sequence are interchanged.

During test, the cell is reconfigured. Although the tasks of all multiplexers and communication lines are interchanged, the functional behaviour of the cell remains exactly as before. For instance, Mux 1 in test mode runs the task of Mux 2 in normal operation and vice versa.

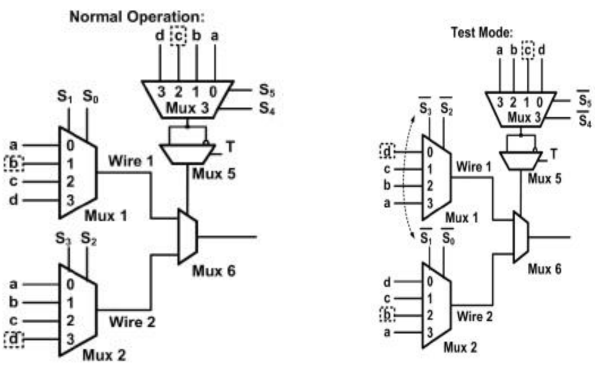

Figure 3: Simplified Core Cell

III. RESULTS AND DISCUSSION

Unitronics Application Utilizing Transposon [Structural Implementation]

Fig.3 shows the core cell of Unitronic architecture which makes use of the Horizontal Gene Transfer characteristic of the prokaryote bacteria. It helps in working fault-less even if the inputs are being flipped due to transient errors. The core cell is being schematically implemented by using Microwind DSCH 2 software. It provides a user-friendly environment for logical design and fast simulation and validation of complex logic structures. Figures [4-7] shows the circuit diagram and respective outputs for both normal mode and test mode.

Fig 4: shows the normal mode operation circuit in the DSCH 2 window. The active lines are red whereas inactive lines are colored blue. Fig 5: shows the output for normal mode circuit. Fig 6: shows the test mode operation circuit and Fig 7: its corresponding output.

Microwind in truly integrated EDA [Electronic Design Automation] software encompassing IC designs from concept to completion, enabling chip designers to design beyond their imagination. Microwind integrates traditionally separated front-end and back-end design into an integrated flow, accelerating the design cycle and reduced design complexities

DSCH [Schematic Editor and Digital Simulator] is a logic editor and simulator. It is used to validate the architecture of the logic circuit before the microelectronics design is started. It provided a user-friendly environment for hierarchical logic design, and fast simulation with delay analysis, which allows the design and validation of complex logic structures.

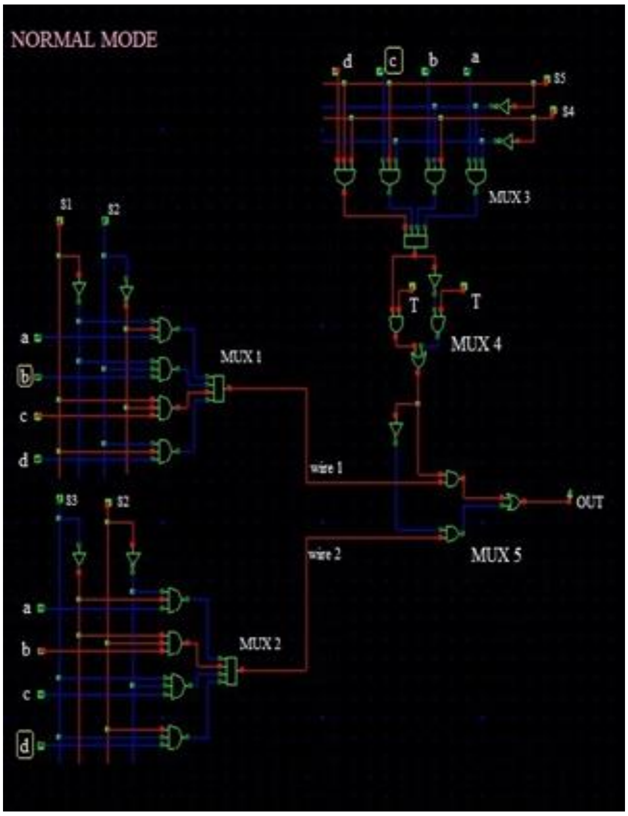

Figure 4: Circuit for normal mode operation

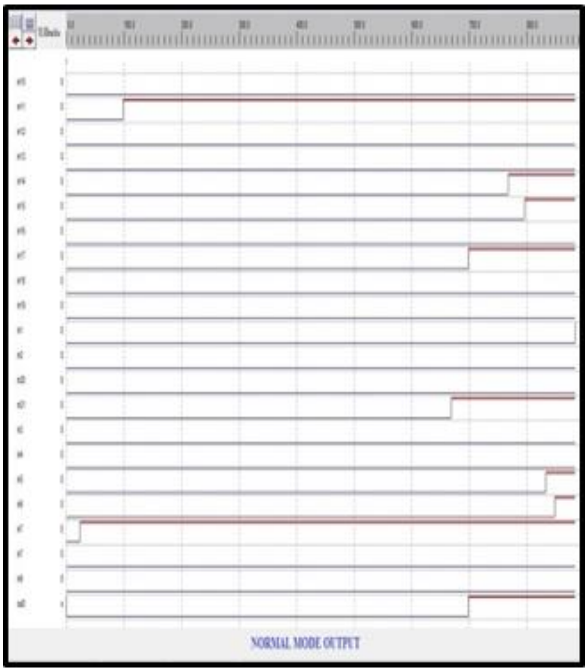

Figure 5: Output of normal mode

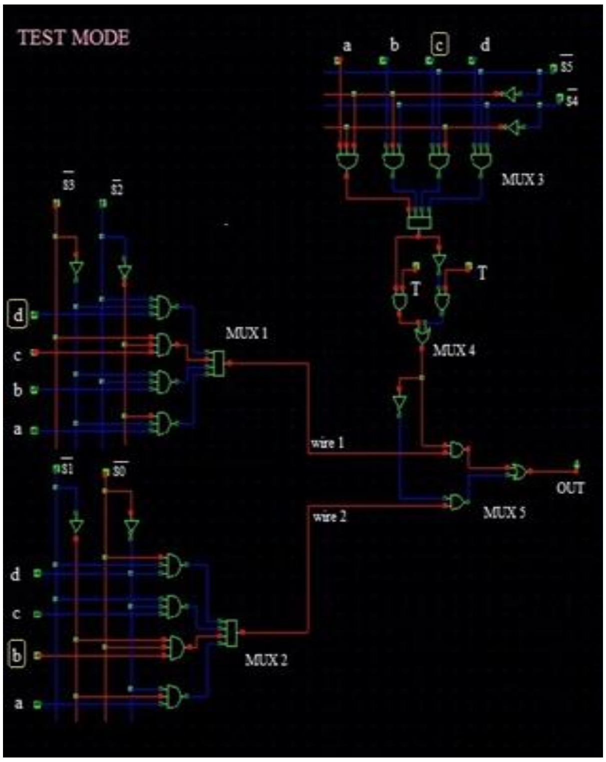

Figure 6: Circuit for test mode operation

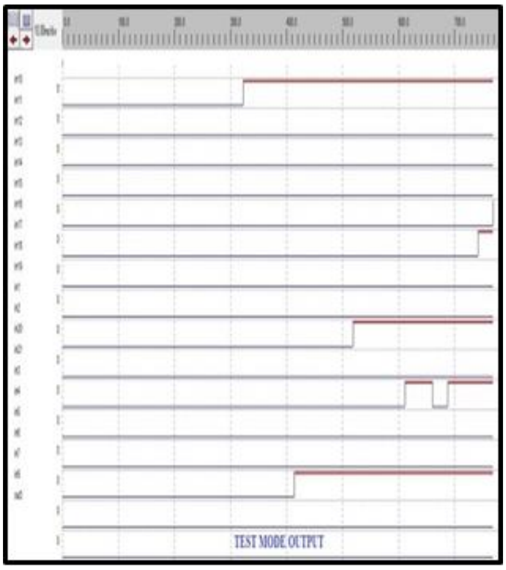

Figure 7: Output of test mode

IV. CONCLUSION

Fault Tolerance is an important characteristic that every system should possess which are employed in life critical applications and in high-cost space applications. In these entire circumstances one cannot afford to have a system to be dead while the mission is still on because it could cause huge losses which cannot be replaced later. It may result either in loss of human life or huge financial loss and efforts of those people behind the project. So it is necessary for such systems to be error-less and the idea that our Mother Nature gives cannot be replaced by anything. And so as the simple prokaryote organisms show that even being smaller and single celled it provides much insight into designing fault tolerant systems that provide much reliability at critical situations. the Horizontal Gene Transfer characteristic of the bacteria employed here, help in designing future systems that could work even when there is change in input data due to the radiations.

V. REFERENCES

[1]. Isaak Yaang, Sung Hoon Jung and Kwang-Hyoon Cho ‘Self-repairing digital system with unified recovery process inspired by endocrine cellular communication’ IEEE transactions on VLSI Systems (June, 2013).

[2]. Daniel mange, Edurado Sanchez, A. Stauffer, G. Tempesti, P. Marchal and c. Piguet ‘Embryonics: A new methodology for designing field programmable gate arrays with self- repair and self-replicating properties ’ (September,2008).

[3]. Edward Stott, Pete Sedcole, Peter y K Cheung ‘Fault Tolerant Methods for reliability in FPGAs’ IEEE Xplore (November 2008).

[4]. NASA Thesaurus- ‘Aerospace science and technology dictionary F section’.

[5]. William c Carter ‘fault tolerant computing: An introduction and viewpoint’ IEEE Transactions on computers (March 1973).

[6]. John Van Neumann ‘theory of self-reproducing automata’ (1966).

[7]. Raymond S Lim ‘Fault tolerant computing’ NASA technical paper.

Author profile

Ancy C P completed her Bachelor’s degree in Electronic and Communication engineering from Vins Christian College of Engineering and presently doing Master’s degree in VLSI Design in Narayanaguru College of engineering, where both the college are affiliated to Anna University, Chennai. She is very much interested in VLSI circuits and its application mainly in space electronic systems.