Diagnosis and Repair Method of SoC Memory

VLADIMIR HAHANOV, ANNA HAHANOVA, SVETLANA CHUMACHENKO, SERGEY GALAGAN
Computer Engineering Faculty
Kharkov National University of Radioelectronics
Lenin Ave. 14, Kharkov, 61166, phone: (057) 70-21-421, (057) 70-21-326
UKRAINE
hahanov@kture.kharkov.ua; kiu@kture.kharkov.ua

Abstract: – An exact method of memory elements diagnosis and repair by spares that enables to cover a set of fault cells by minimally possible quantity of spares is represented. The method is oriented on implementation to the Infrastructure Intellectual Property for SoC functionality. It enables to raise yield essentially on the electronic technology market by means of faulty chip repair in the process of production and operation, as well as to increase the life cycle duration of memory matrixes by repair of them in real time.

Key-Words: - Diagnosis, System-on-Chip, Built-In Repair Analysis, Built-In Self Repair, Fault, Simulation.

1 SOC Infrastructure Intellectual Property Technologies

Computational and hardware complexity of modern digital systems on a chip (SoC) is characterized by millions of equivalent gates and requires making and implementation of new high-level design technologies: Electronic System Level (ESL) Design, Transaction Level Modeling (TLM) and embedded service – Infrastructure Intellectual Property (I-IP). It means that search for high-performance methods and facilities [1-17, 19-21] reduces all researchers to necessity to rise of an abstraction level of Functional Intellectual Property (F-IP) models, which are created and embedded into a chip. EDA market suggests facilities for computer-aided modelling and verification of system level devices, beginning with HDL-compilers (C++, SystemC, SystemVerilog, UML, SDL) [12] up to graphics environments (Simulink, LabView, Xilinx EDK). These facilities enable to create projects using existing library components by means of ESL-mapping and creation of TLM-interfaces [13, 14]. Market attractiveness of a digital system implementation to FPGA is determined by the followings: application of relatively cheap chips instead of the universal processors, low power consumption, small overall size, qualitative and reliable realization of the main functions due to on-chip I-IP-infrastructure that is urgent in the century of mobile computers.

The memory diagnosis and repair problem is related to the tendency to continuous reduction of chip area, which is allocated to original and standardize logic, and simultaneous growth of embedded memory. As it is represented in Fig. 1, increase of the memory specific weight on a chip reduces to its complete dominance for data and program storage, which will reach 94% by 2014 year [2]. It will provide not only fast response of carrying out of functions, but also flexibility that is appropriate to software in relation to design error correction.

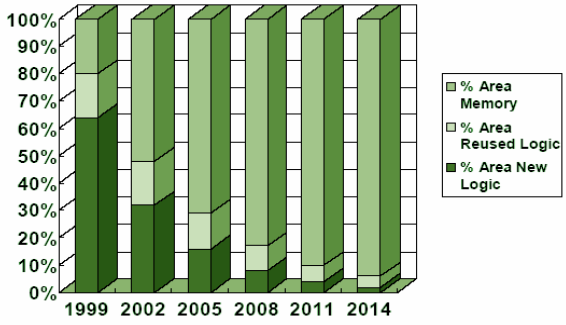


Fig. 1. Memory specific weight on a chip SoC

The memory element feature is the fact that some cells can fail under fault influence in the process of production and operation. This fact not always brings a memory matrix into a critical state, when the repair is not possible. So, such technical memory state, when the total quantity of faulty cells is not greater than spare capacities of a device (intended for repair), is considered below.

The research aim is development of algebra-logical method of embedded matrix memory diagnosis and repair in real time.

The problems: 1) SoC Infrastructure Intellectual Property Technologies; 2) An Infrastructure Intellectual Property method on basis of the covering matrix; 3) Formalization of the algebra-logical AL-method for embedded memory repair; 4) Practical results.

Modern design technologies of digital systems on chips propose along with creation of functional blocks F-IP development of service modules I-IP, which are oriented on complex solving of the project quality problem and yield increasing in manufacturing that is determined by implementation of the following services into a chip [13]:

1) Diagnosis of failures and faults by analysis of information, which is obtained on the stage of testing and use of special methods of embedded fault lookup on basis of the standard IEEE 1500 [9,15,17];

2) Repair of functional modules and memory after fixation of a negative testing result, fault location and identification of a fault type in carrying out of the diagnosis phase;

3) Measurement of the general characteristics and parameters of a device operation on basis of on-chip facilities, which enable to make time and volt-ampere measurements;

4) Reliability and fault tolerance of a device operation in working that is obtained by diversification of functional blocks, redundancy of them and repair of SoC in real time.

2 Memory Diagnosis and Repair Method

It is represented the exact method of memory elements diagnosis and repair by spares that enables to cover a set of fault cells by minimally possible quantity of spares. The method is oriented on implementation to the Infrastructure Intellectual Property for SoC functionality. The structure solutions for realization of the method of diagnosis and repair of memory matrix fault cells are proposed. [6-9, 14, 18].

In the process of operation and repair any kinds of memory it is necessary the guarantee of its technical compliance. For this the carrying out of three procedures is provided: 1) Memory testing that consists of test patters input, which oriented on identification of specific kinds of faults [1, 8]; 2) In the case of fail appearance it is necessary an additional diagnosis procedure that enables to determine location, cause and kind of a fault; 3) After detection of a fault set, which block carrying out of the memory function, it is necessary to activate the repair process – replacement of faulty elements by spares, which initially are on a chip [6,7]. Thereby, aforementioned actions are oriented on the growth of yield without significant additional time and material costs. To repair it is necessary a special mechanism of memory repair by means of replacement of faulty components by faultless ones from chip reserve.

As a rule the testing procedure is realized by BIST-block (Built-In Self Test), which is hardware fast-acting generator of test patterns, as well as an analyzer (signature) of reactions of memory outputs on test patterns. Repair analysis consists of definition of covering possibility of faulty memory elements by available reserve components. Memory module has two parts: 1) functional cells, which are used for data and program storage, when a module is used in SoC; 2) reserve or spare cells, which are designed for memory repair in case of functional cells failure. Functional and reserve cells are joined to columns and rows. When a fault is detected, a row (a column), which includes a faulty element, is disconnected from the functional structure of memory cells and a row (a column) from chip reserve is connected on its place. A number of reserve components is limited, so it is necessary a special mechanism of effective allocation of repair resource for support of faulty memory elements covering by minimally possible quantity of redundant rows and columns.

The search procedure of faulty cells covering by minimal quantity of reserve rows and columns described above can be realized as on-chip repair module or external one. In second case data about errors is received from external modules, they are processed and pass to the controller that provides memory repair. It results in considerable time loss. So, the preferable solution is on-chip module realization, when data about errors is passed from BIST directly. Such mechanism is called as BIRA [6, 11] – Built-In Repair Analysis.

Memory repair is realized by disconnection of faulty elements (rows and columns of a matrix) by means of electrical fusion of metal links and connection of reserve ones. The fuse process can be electrical or laser. Electrical fuse equipment has smaller dimensions then laser one and it is used more frequently. Fuse is carried out by means of an instruction set, which can store in permanent memory inside chip (hard repair) or in random-access memory (soft repair) [6-8]. Soft repair has several advantages: when a defect appears a new corrected instruction can be recorded to memory easily; there provide economic use of chip area and sufficient reliability [4]. Hard repair enables to use a simplified manufacturing test and provides detection of errors, which can not be fixed by soft repair under certain circumstances (for instance, overheat).

The structure of on-chip memory analysis and soft repair processes – BISR – [6-8] is represented in Fig. 2: 1) Chip activation, filling of the BISR register by zero values. 2) Run the BIST controller. Memory testing and accumulation of information about faulty cells in the BIRA register. 3) Transfer of information about faulty cells to the BISR register for subsequent fusion. 4) Scanning the BIRA registers, which contain the repair status, by the BIST controller for obtainment of faults information. 5) Run the fuse controller in record mode and transfer the repair instructions from the BISR. 6) Chip restart. Recording the fuse information to the BISR register, replacement of faulty rows and columns by reserve components is fulfilled. 7) Run the BIST controller for repeated memory testing and verification of the repair result correctness.

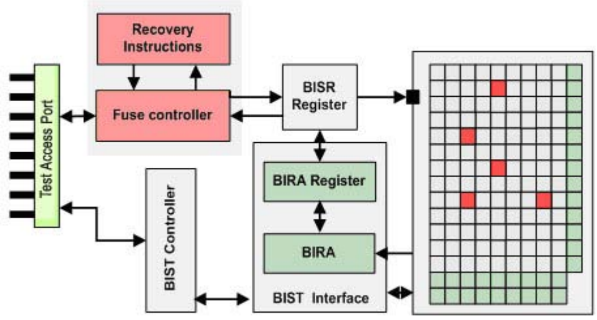


Fig. 2. Flow of on-chip memory analysis and repair

The aim function Z of given research can be defined on the basis of modern progress in the field of on-line memory repair in the following way: minimization of the repair cost (hardware costs) of a memory module $M = |M_{ij}|$ in the process of SoC operation by means of use the algebra-logical method of minimization of the faulty cells set covering by a system of reserve elements under the constraints N on quantity of ones:

$$Z = \min_i [Q_i(F)]_{|Q_i(F) \le N_{max} = N_r + N_c},$$

where $Q_i(F)$ – the cost of i-th solution variant of the memory module $M = |M_{ij}|$ repair by means of the minimal subset of rows and columns $R = \{R_r, R_c\}$ of chip reserve that covers the set F of faulty memory cells $R \cap F = F, Z^* = \max |F_i|, F_i \in F \leftarrow \forall R_i$.

Method of minimal covering obtainment on an example a memory matrix with five faulty cells [11], two reserve rows and a reserve column (Fig. 3) is considered below. Every reserve component (a row or a column) can repair from one up to n faulty cells, which belong to a row or a column.

The method idea is optimal replacement of faulty memory matrix elements by means of solution of the covering problem of faulty columns by row reserve. For the method illustration it is proposed originally to use the covering matrix of given faults F by some quantity of rows (it can be test patterns or reserve rows) X and

$$|F| \ge |X| = \{F_1, F_2, F_3, F_4, F_5, F_6, F_7, F_8\} \ge$$
$$\ge \{X_1, X_2, X_3, X_4, X_5, X_6\}.$$

Let the matrix Y is specified:

$$(F_j \cap X_i \ne \varnothing \rightarrow Y_{ij} = 1) \& (F_j \cap X_i = \varnothing \rightarrow Y_{ij} = 0):$$

Y =

X_i / F_j	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8
X_1	1					1		
X_2		1	1				1	1
X_3					1			1
X_4	1			1				
X_5			1	1				
X_6					1	1		

The exact solution of the covering problem of faults by minimal quantity of reserve memory rows is based on synthesis of the Boolean function that is written as product of sums, written by constituents of unities, which correspond to columns of the matrix:

$$Y = (X_1 \vee X_4) \& (X_2) \& (X_2 \vee X_5) \& (X_4 \vee X_5) \&$$
$$\& (X_3 \vee X_6) \& (X_1 \vee X_6) \& (X_2) \& (X_2 \vee X_3).$$

In given case an analytic notation in the Boolean function form, represented in conjunctive normal form (CNF), is the initial model that contains a full set of covering problem solutions that is solved by finding of disjunctive normal form (DNF). The transformation procedure of CNF to DNF by means of all terms multiplication is performed for it. In result of the equivalent transformations, performed in compliance with the algebra of logic rules, it is came out the Boolean function that contains all possible fault covers, defined by four variants of row combinations:

$$Y = (X_1X_2 \vee X_2X_4)(X_2X_4 \vee X_4X_5 \vee X_5X_5 \vee X_2X_5) \&$$
$$\& (X_1X_3 \vee X_1X_6 \vee X_6X_6 \vee X_3X_6)(X_2X_2 \vee X_2X_3) =$$
$$= (X_1X_2X_3X_4 \vee X_2X_4X_6 \vee X_1X_2X_3X_5 \vee X_1X_2X_5X_6).$$

The minimal solution of covering problem contains three reserve rows, which can cover 8 faults in a memory matrix: $Y = X_2X_4X_6$.

For use of the proposed memory repair method it is necessary to remember that every fault F_i in a memory matrix belongs to a row and a column simultaneously. So, transformation of the topological fault model to the covering matrix consists of assignment of row and column numbers, which are distorted by given fault, to every fault.

For instance (Fig.3), where there are 5 faulty cells, which are covered by three columns and 4 rows, the transformation turns a memory matrix into the covering table, where left column specifies one-to-one correspondence between fault coordinates (row and column numbers of a memory matrix) and rows of a fault covering:

$Y =$

X_i / F_j	F_1	F_2	F_3	F_4	F_5
$C_2 \to X_1$		1		1	
$C_4 \to X_2$			1		1
$C_8 \to X_3$	1				
$R_3 \to X_4$	1	1			
$R_5 \to X_5$			1		
$R_7 \to X_6$				1	
$R_{10} \to X_7$					1

In other words, the memory matrix topology is transformed from two-dimensional metrics to one-dimensional row structure, which have defined covering features concerning about fault columns.

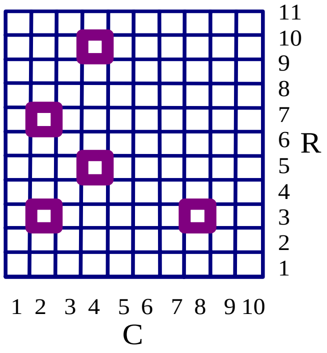


Fig. 3. Memory matrix with fault cells

The following Boolean function forms logical product of disjunctions, written by constituents of unities, corresponding to columns of the matrix $(F_j \cap X_i \neq \varnothing \to Y_{ij} = 1)$:

$$Y = (X_3 \vee X_4)(X_1 \vee X_4)(X_2 \vee X_5)(X_1 \vee X_6)(X_2 \vee X_7) =$$
$$= (X_1X_3 \vee X_1X_4 \vee X_3X_4 \vee X_4)(X_1X_2 \vee X_1X_5 \vee X_2X_6 \vee$$
$$\vee X_5X_6)(X_2 \vee X_7) = (X_1X_3 \vee X_4)(X_2 \vee X_7)(X_1X_2 \vee$$
$$\vee X_1X_5 \vee X_2X_6 \vee X_5X_6) = (X_1X_2X_3 \vee X_2X_4 \vee X_1X_3X_7 \vee$$
$$\vee X_4X_7)(X_1X_2 \vee X_1X_5 \vee X_2X_6 \vee X_5X_6) = (X_1X_2X_3 \vee$$
$$\vee X_1X_2X_4 \vee X_1X_2X_3X_7 \vee X_1X_2X_4X_7 \vee X_1X_2X_3X_5 \vee$$
$$\vee X_1X_2X_4X_7 \vee X_2X_4X_6 \vee X_2X_4X_5X_6 \vee X_1X_2X_3X_7 \vee$$
$$\vee X_1X_3X_5X_7 \vee X_1X_2X_3X_6X_7 \vee X_1X_3X_5X_6X_7 \vee$$
$$\vee X_1X_2X_4X_7 \vee X_1X_4X_5X_7 \vee X_2X_4X_6X_7 \vee X_4X_5X_6X_7 =$$
$$= (X_1X_2X_3 \vee X_1X_2X_4 \vee X_2X_4X_6 \vee X_1X_3X_5X_7 \vee$$
$$\vee X_1X_4X_5X_7 \vee X_4X_5X_6X_7.$$

The equivalent transformations enable to simplify the complex construction – conjunctive normal form – and to obtain the minimal set of all solutions, a number of that is equal to six in this case. Subset of minimal solutions is defined by three conjunctive terms, every of which contains 3 reserve elements for memory matrix repair:

$$Y = X_1X_2X_3 \vee X_1X_2X_4 \vee X_2X_4X_6.$$

3 Formalization of Algebra-Logical Memory Repair Method

The aim function is defined as minimization of reserve components of the memory matrix (S – spare), which are needed for its repair in the process of SoC operation by means of synthesis of disjunctive normal form of faulty elements covering and subsequent choice of the minimal conjunctive term $X^t(R^t, C^t) \in Y$ that satisfies to limitations on quantity of the reserve rows and columns S^r_{max}, S^c_{max}, which enter into the logical product:

$$Z = \underset{t=1,n}{\min}(|X^t|)\Big|_{|S^r|+|S^c| \le S_{max};\, |S^r| \le S^r_{max};\, |S^c| \le S^c_{max}},$$

$$X^t \in Y = \{X^1, X^2, ..., X^t, ..., X^n\},$$

$$X^t = (X_1^t \,\&\, X_2^t \&, ..., \& X_i^t \&, ..., \& X_{m_t}^t),$$

where every resulting conjunctive term of the function Y is made from the row and column identifiers $X^t = (R^t, C^t)$, which cover all faults in a memory matrix. The best solution is a term of minimal length at Quine mark [10], in which there are rows and columns, covering all faults. In particular case a solution can contain none rows (columns), when existing columns (rows) from memory matrix reserve are sufficient for memory repair. The model of definition process of minimal quantity of spares, which cover all detected faults in a memory matrix, comes to the following items:

1. Transformation of two-dimensional model of a memory matrix faults to the fault covering table by reserve rows and columns. To achieve of the aim the topological memory model in the form of matrix, identifying detected faults, is considered:

$$M = |M_{ij}|, M_{ij} = \begin{cases} 1 \leftarrow T \oplus f = 1; \\ 0 \leftarrow T \oplus f = 0. \end{cases}$$

Here a matrix coordinate is equal to 1, if the fault-free behaviour function of a cell gives unit value on a test, the coordinate is identified as faulty. After fixation of all faults construction of the fault covering table $Y = \left|Y_{ij}\right|, i = \overline{1,n}; j = \overline{1,m}$ is carried out, where columns correspond to the set of detected faults m and rows are numbers of columns and rows of a memory matrix, which have faults:

$$Y = \left|Y_{ij}\right|, Y_{ij} = \begin{cases} 1 \leftarrow C_i(R_i) \cap F_j \neq \varnothing; \\ 0 \leftarrow C_i(R_i) \cap F_j = \varnothing. \end{cases}$$

Instead of the two-dimensional metrics components C and R the one-dimensional vector is used, it is concatenated from two sequences C and R, the power of which is equal to n = p + q:

$$X = C*R = (C_1, C_2, ..., C_i, ..., C_p)*(R_1, R_2, ..., R_j, ..., R_q) =$$
$$= X^c*X^r = (X_1, X_2, ..., X_i, ..., X_p, X_{p+1}, X_{p+2}, ...,$$
$$X_{p+j}, ..., X_{p+q}).$$

At that there exists one-to-one correspondence between the initial set elements (C, R) and the resulting vector X, which is defined in first column of the matrix Y. It is necessary to say that transformation $X = C*R$ is carried out for ease of consideration and subsequent forming of disjunctive normal form (uniformity of variables, forming the Boolean function). If the procedure is not carried out the function is defined by two kinds of variables, containing rows and columns of a memory matrix..

2. Construction of conjunctive normal form for analytic, complete and exact solution of the covering problem. After generation of the covering matrix that contains zero and unit coordinates the synthesis of analytic covering form is carried out by writing of CNF by columns. Here a number of conjunctive terms are equal to quantity of table columns and every disjunction is written by unit values of a current column:

$$Y = \mathop{\wedge}_{j=1}^{m} (Y_{pj} \vee Y_{qj})_{\{Y_{pj}, Y_{qj}\}=1} = \mathop{\wedge}_{j=1}^{m} (X_{pj} \vee X_{qj}).$$

From last expression it is obvious that every column has two coordinates only, which have unit value, and a number of logical products is equal to total quantity of faults m, detected in a memory matrix.

3. Transformation of CNF to DNF that enables to determine all solutions of the covering problem. For that it is necessary to apply an operation of logical multiplication and the minimization (absorption) rules to conjunctive normal form to obtain of disjunctive normal form:

$$Y = \mathop{\vee}_{j=1}^{w} (k_1^j X_1 \wedge k_2^j X_2 \wedge ... \wedge k_i^j X_i \wedge ... \wedge k_n^j X_n), k_i^j = \{0,1\}.$$

It is the generalized DNF notation, where in the limit a number of terms is equal to $w = 2^n$, where n is quantity of rows in the generalized set (C,R) or quantity of the variables X in the matrix Y, on the set of which all solutions are formed (fault covering by reserve components); if k_i^j at X_i is equal to zero the variable X_i is nonessential.

4. Choice of minimal and exact solutions of the covering problem. It is related to determination of minimal length conjunctions in the obtained DNF. The following transformation executing rows and columns of a memory matrix on basis of above-mentioned correspondence enables to write a minimal covering or set of ones in two-dimensional metrics of rows and columns, which satisfies the conditions (limitations) of the aim function on quantity of reserve components.

An illustration of the memory matrix repair process model is proposed below. Minimal quantity of reserve components, which cover all faults, is determined. A memory matrix with faults and reserve [11] is represented in Fig. 4.

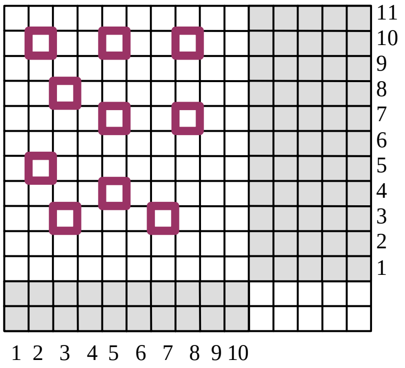


Fig. 4. Memory matrix with faults and reserve

The matrix has limitations on diagnosis and repair capabilities of ten faulty cells, which are defined by two rows and five columns. In compliance with item 1 of the model of definition process of minimal quantity of spares, which covers all detected faults in a memory matrix, the covering table of ten faults

$$F = (F_1, F_2, F_3, F_4, F_5, F_6, F_7, F_8, F_9, F_{10})$$

is formed. Faults are covered by 11 rows, which are represented in the form of concatenation of the subsets C and R, which are in one-to-one correspondence with the variable vector X:

$$C*R = (C_2, C_3, C_5, C_7, C_8)*(R_3, R_4, R_5, R_7, R_8, R_{10}) \approx$$
$$\approx X = (X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8, X_9, X_{10}, X_{11}).$$

Y =

X_i / F_j	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}
$C_2 \to X_1$				1						1
$C_3 \to X_2$		1					1			
$C_5 \to X_3$			1			1			1	
$C_7 \to X_4$	1									
$C_8 \to X_5$					1			1		
$R_3 \to X_6$	1	1								
$R_4 \to X_7$			1							
$R_5 \to X_8$				1						
$R_7 \to X_9$					1	1				
$R_8 \to X_{10}$							1			
$R_{10} \to X_{11}$								1	1	1

In compliance with the covering table construction of DNF is performed, the terms are written by unit values of columns:

$$Y = (X_4 \vee X_6)(X_2 \vee X_6)(X_3 \vee X_7)(X_1 \vee X_8)(X_5 \vee X_9)\,\&$$

$$\&(X_3 \vee X_9)(X_2 \vee X_{10})(X_5 \vee X_{11})(X_3 \vee X_{11})(X_1 \vee X_{11}).$$

Subsequent transformations related to obtainment of disjunctive normal form are based on application of The Boolean algebra lows and identities, which enable to carry out logical multiplication of all ten multipliers, subsequent minimization of DNF terms by application of the minimization operator, the absorption axioms, removal of the same terms. Skipped intermediate calculus the final result is represented in the following form:

$$\begin{aligned}
Y = {} & X_1 X_2 X_3 X_4 X_5 \vee X_2 X_3 X_4 X_5 X_8 X_{11} \vee \\
& \vee X_1 X_2 X_4 X_9 X_3 X_{11} \vee X_1 X_3 X_2 X_4 X_9 X_{10} X_{11} \vee \\
& \vee X_1 X_7 X_{10} X_{11} X_6 X_9 \vee X_6 X_9 X_7 X_8 X_{10} X_{11} \vee \\
& \vee X_2 X_4 X_9 X_3 X_8 X_{11} \vee X_1 X_2 X_4 X_9 X_7 X_{11} \vee \\
& \vee X_2 X_4 X_9 X_7 X_8 X_{11} \vee X_3 X_2 X_4 X_9 X_8 X_{10} X_{11} \vee \\
& \vee X_1 X_2 X_4 X_9 X_7 X_{10} X_{11} \vee X_1 X_2 X_3 X_5 X_6 \vee \\
& \vee X_1 X_3 X_5 X_6 X_{10} \vee X_2 X_3 X_5 X_6 X_8 X_{11} \vee \\
& \vee X_3 X_5 X_6 X_8 X_{10} X_{11} \vee X_1 X_2 X_3 X_{11} X_6 X_9 \vee \\
& \vee X_1 X_3 X_{10} X_{11} X_6 X_9 \vee X_2 X_3 X_8 X_{11} X_6 X_9 \vee \\
& \vee X_1 X_2 X_7 X_{11} X_6 X_9 \vee X_2 X_7 X_8 X_{11} X_6 X_9 \vee \\
& \vee X_3 X_8 X_{10} X_{11} X_6 X_9 .
\end{aligned}$$

The choice of minimal length terms, which contain 5 variables, forms a set of optimal (minimal) solutions:

$$Y = X_1X_2X_3X_4X_5 \vee X_1X_2X_3X_5X_6 \vee X_1X_3X_5X_6X_{10}.$$

Transformation of obtained function to a coverage that contains variable designations in the form of rows and columns of a memory matrix enables to represent solutions in the following form:

$$Y = C_2C_3C_5C_7C_8 \vee C_2C_3C_5C_8R_3 \vee C_2C_5C_8R_3R_8.$$

All obtained minimal solutions satisfy the requirements (limitations) on spare quantity that is determined by numbers:

$$(\left|C^r\right| \le 5)\,\&\,(\left|R^r\right| \le 2).$$

Other solutions, determined in DNF, are no interest, because they have not optimal covering of faulty cells that is determined by quantity of variables (rows + columns) in terms, greater then five. Subsequent technology of embedded repair of faulty cells consists of electrical reprogramming of an address decoder of a column or a row of a memory matrix. In respect to memory, represented in Fig. 4, a procedure of writing or reading of information at access to any cell of column 2 will be readdressed to reserve column 11. In compliance with last obtained solution (first term of DNF function Y) other faulty columns will be replaced on fault-free ones from memory reserve: 3 – on 12; 5 – on 13; 7 – on 14, 8 – on 15.

The computational complexity of algebra-logical memory repair method in the part of solving of the covering problem is determined by the following expression:

$$Q = 2^{|F|} + |C + R| \times 2^{|F|},$$

where $2^{|F|}$ is costs related to DNF synthesis by logical multiplication of two-component disjunctions (fault coordinate is defined by row and column numbers), quantity of them is equal to quantity of faulty cells; $|C + R| \times 2^{|F|}$ is upper limit of computational costs, which are needed for minimization of obtained DNF on maximum set of variables that is equal to total quantity of rows and columns $|C + R|$. In the worst case, when coordinates of all faulty cells are not correlated by rows and columns (they are unique), for instance, diagonal faults, the computational complexity of the matrix method is dependent from quantity of faulty cells only and its analytic notation is transformed to the following view:

$$Q = 2^{|F|} + |C + R| \times 2^{|F|}\Big|_{|C+R| \le 2\times|F|} =$$

$$= 2^{|F|} + 2 \times |F| \times 2^{|F|} = 2^{|F|} \times (1 + 2 \times |F|)$$

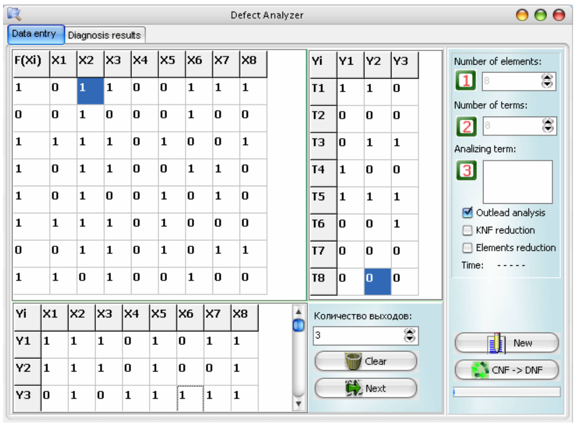
If instead of fault set power to use quantity m of them, the previous expression can be represented in more simple form:

$$Q = 2^m \times (1 + 2 \times m) = 2^m (2m + 1).$$

According to the SoC Functional Intellectual Property Infrastructure, the matrix repair method on basis of solving of the covering problem is implemented into a chip as one of I-IP components that is designed for the operability support of SoC matrix memory.

4 Software «Defect Analyzer»

«Defect Analyzer» is designed for simulation of SoC testing and diagnosis by using the fault detection table. It is application that functions under the operating system Windows XP. The algebra-logical diagnosis method, successive and parallel methods of the test point choice are realized in the application. The input data is the covering table that is formed in the testing process of a circuit. The table shows input signal levels, as well as it displays faults in a circuit on current test pattern (value of the vector V is a sign that, Fig. 5).



Defect Analyzer

Data entry | Diagnosis results

F(Xi)	X1	X2	X3	X4	X5	X6	X7	X8
1	0	1	1	0	0	1	1	1
0	0	1	0	0	0	1	0	0
1	1	1	1	0	1	0	0	1
1	0	1	1	0	0	1	1	0
1	0	1	0	0	1	0	1	0
1	1	1	1	0	1	0	0	0
0	0	1	1	0	1	0	1	1
1	1	0	1	0	0	1	0	0

Yi	Y1	Y2	Y3
T1	1	1	0
T2	0	0	0
T3	0	1	1
T4	1	0	0
T5	1	1	1
T6	0	0	1
T7	0	0	0
T8	0	0	0

Number of elements: 1 8

Number of terms: 2 8

Analizing term: 3

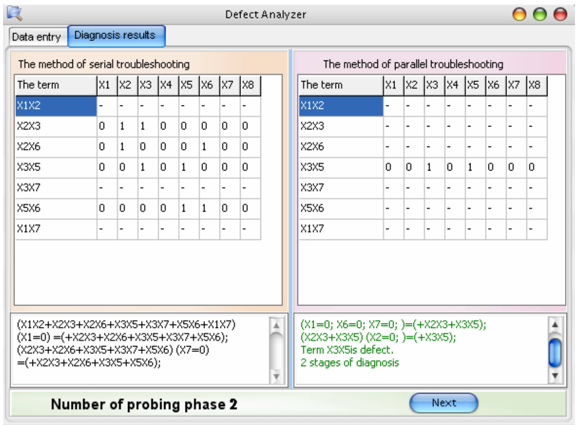
☑ Outlead analysis
☐ KNF reduction
☐ Elements reduction
Time: - - - - -

Yi	X1	X2	X3	X4	X5	X6	X7	X8
Y1	1	1	1	0	1	0	1	1
Y2	1	1	1	0	1	0	0	1
Y3	0	1	0	1	1	1	1	1

Количество выходов: 3

Fig. 5. The main window of «Defect Analyzer»

To simplify the initial covering table information about zero elements of test patterns and output signal levels for every test pattern can be used. Also reduction CNF to more simple form can be performed. Probing procedure is realized step by step in specified test points. It is displayed in the form of covering table and simplified DNF (Fig. 6).



Defect Analyzer

Data entry | Diagnosis results

The method of serial troubleshooting

The term	X1	X2	X3	X4	X5	X6	X7	X8
X1X2	-	-	-	-	-	-	-	-
X2X3	0	1	1	0	0	0	0	0
X2X6	0	1	0	0	0	1	0	0
X3X5	0	0	1	0	1	0	0	0
X3X7	-	-	-	-	-	-	-	-
X5X6	0	0	0	0	1	1	0	0
X1X7	-	-	-	-	-	-	-	-

(X1X2+X2X3+X2X6+X3X5+X3X7+X5X6+X1X7)
(X1=0) =(+X2X3+X2X6+X3X5+X3X7+X5X6);
(X2X3+X2X6+X3X5+X3X7+X5X6) (X7=0)
=(+X2X3+X2X6+X3X5+X5X6);

The method of parallel troubleshooting

The term	X1	X2	X3	X4	X5	X6	X7	X8
X1X2	-	-	-	-	-	-	-	-
X2X3	-	-	-	-	-	-	-	-
X2X6	-	-	-	-	-	-	-	-
X3X5	0	0	1	0	1	0	0	0
X3X7	-	-	-	-	-	-	-	-
X5X6	-	-	-	-	-	-	-	-
X1X7	-	-	-	-	-	-	-	-

(X1=0; X6=0; X7=0;)=(+X2X3+X3X5);
(X2X3+X3X5) (X2=0;)=(+X3X5);
Term X3X5is defect.
2 stages of diagnosis

Number of probing phase 2

Fig. 6. Probing procedure

Data level, which can be processed by the application, is limited by quantity of terms of no simplified DNF. A number of table rows (terms) should not be greater then 32766. A number of DNF terms depends on initial data and it can be calculated by formula:

$$N = \underset{\forall i(V_i=1)}{\overset{i=\overline{1,n}}{P}} (N_i),$$

where Ni is quantity of elements "1" in a test pattern. At diagnosis the input patterns for which value of the vector V is equal to 1 is considered. The main time of diagnosis is consumed on transformation CNF to DNF and its reduction. Time required for the transformation depends on experiment result, circuit dimension and it is in proportion to the coefficient k that is calculated by formula:

$$k = k_1 \underset{i=1}{\overset{n}{P}} ((N_i!)) + k_2 \frac{n!}{2(n-2)!} nm,$$

where n is quantity of test patterns; m is a number of elements, which are used in an experiment; Ni is a number of "1" elements in i-th input pattern; k1 and k2 are coefficients. Time of transformation CNF to DNF versus quantity of test patterns and elements is presented in Fig. 7.

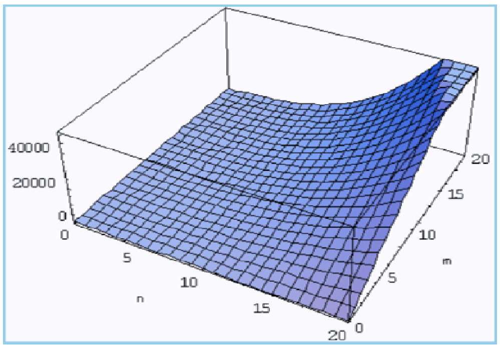


Fig. 7. Time of transformation CNF to DNF

Information about zero elements, reduction of the fault covering table, transformation the initial CNF to more simple form before its transformation to DNF enable to decrease the diagnosis time essentially.

An example. A circuit, represented in Fig. 8, is given. A fault detection table (it includes the first 5 rows of the table below) and an experimental validation vector V correspond to it.

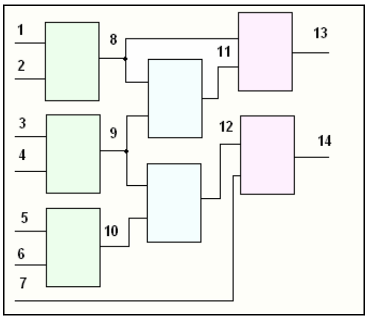


Fig. 8. A circuit example

T_i / F_j	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	V	Y_1	Y_2
T_1		1		1			1								0		
T_2	1					1		1		1	1		1	1	1	1	
T_3			1				1								0		
T_4	1	1	1	1			1	1	1		1			1	1		1
T_5	1		1		1		1		1	1		1	1		1	1	1
Y_1	1	1	1	1				1	1		1		1				
Y_2			1	1	1	1	1		1	1		1		1			

Masking of FDT rows 1,2,4,5 by the vectors $\vec{Y}_1, \vec{Y}_2$, which are represented in bottom of the table, results in decrease of "1" coordinates quantity in FDT:

T_i / F_j	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	V
T_1		1		1			1								0
T_2	1					0		1		0	1		1		1
T_3			1				1								0
T_4	0	0	1	1			1	0	1		0			1	1
T_5	1		1		1		1		1	1		1	1		1

Then the column numbers where zero values of the vector V correspond "1" values of the table row coordinates are determined; the columns (for the example they are F2, F3, F4, F7) are eliminated:

T_i / F_j	F_1	F_5	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}
T_2	1		1			1		1	
T_4				1					1
T_5	1	1		1	1		1	1	

CNF is constructed by "1" values of the vector V:

$$F=(F_1 \vee F_8 \vee F_{11} \vee F_{13})(F_9 \vee F_{14})(F_1 \vee F_5 \vee F_9 \vee F_{10} \vee$$
$$\vee F_{12} \vee F_{13}) = F_1F_9 \vee F_1F_{14} \vee F_8F_9 \vee F_5F_8F_{14} \vee$$
$$\vee F_8F_{10}F_{14} \vee F_8F_{12}F_{14} \vee F_9F_{11} \vee F_5F_{11}F_{14} \vee$$
$$\vee F_{10}F_{11}F_{14} \vee F_{11}F_{12}F_{14} \vee F_9F_{13} \vee F_{13}F_{14}.$$

So, there is the multiple fault F_8F_9 in a circuit. Successive probing in test points is realized as described below.

F_i^1	F_1	F_5	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}.
$W(F_i)$	2	2	4	4	2	4	2	2	8

In the instance the minimal weight of elements F_i is equal to 2. To prove the existence of multiple fault F_8F_9 it is necessary to choose point F_1 as first test point:

$$F_1F_9 \vee F_1F_{14} \vee F_8F_9 \vee F_5F_8F_{14} \vee F_8F_{10}F_{14} \vee F_8F_{12}F_{14} \vee$$
$$\vee F_9F_{11} \vee F_5F_{11}F_{14} \vee F_{10}F_{11}F_{14} \vee F_{11}F_{12}F_{14} \vee F_9F_{13} \vee$$
$$\vee F_{13}F_{14}(F_1 = 0) = F_8F_9 \vee F_5F_8F_{14} \vee F_8F_{10}F_{14} \vee$$
$$\vee F_8F_{12}F_{14} \vee F_9F_{11} \vee F_5F_{11}F_{14} \vee F_{10}F_{11}F_{14} \vee$$
$$\vee F_{11}F_{12}F_{14} \vee F_9F_{13} \vee F_{13}F_{14};$$

F_i^1	F_1	F_5	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14};
$W(F_i)$	0	2	4	3	2	4	2	2	7

$$F_8F_9 \vee F_5F_8F_{14} \vee F_8F_{10}F_{14} \vee F_8F_{12}F_{14} \vee F_9F_{11} \vee F_5F_{11}F_{14} \vee$$
$$\vee F_{10}F_{11}F_{14} \vee F_{11}F_{12}F_{14} \vee F_9F_{13} \vee F_{13}F_{14(F_5=0)} =$$
$$= F_8F_9 \vee F_8F_{10}F_{14} \vee F_8F_{12}F_{14} \vee F_9F_{11} \vee F_{10}F_{11}F_{14} \vee$$
$$\vee F_{11}F_{12}F_{14} \vee F_9F_{13} \vee F_{13}F_{14};$$

F_i^1	F_1	F_5	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14};
$W(F_i)$	0	0	3	3	2	3	2	2	5

$$F_8F_9 \vee F_8F_{10}F_{14} \vee F_8F_{12}F_{14} \vee F_9F_{11} \vee F_{10}F_{11}F_{14} \vee$$
$$\vee F_{11}F_{12}F_{14} \vee F_9F_{13} \vee F_{13}F_{14(F_{13}=0)} =$$
$$= F_8F_9 \vee F_8F_{10}F_{14} \vee F_8F_{12}F_{14} \vee F_9F_{11} \vee$$
$$\vee F_{10}F_{11}F_{14} \vee F_{11}F_{12}F_{14};$$

F_i^1	F_1	F_5	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14};
$W(F_i)$	0	0	3	2	2	3	2	0	4

$$F_8F_9 \vee F_8F_{10}F_{14} \vee F_8F_{12}F_{14} \vee F_9F_{11} \vee F_{10}F_{11}F_{14} \vee$$
$$\vee F_{11}F_{12}F_{14(F_{12}=0)} = F_8F_9 \vee F_8F_{10}F_{14} \vee$$
$$\vee F_9F_{11} \vee F_{10}F_{11}F_{14};$$

F_i^1	F_1	F_5	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14};
$W(F_i)$	0	0	2	2	2	2	0	0	2

$$F_8F_9 \vee F_8F_{10}F_{14} \vee F_9F_{11} \vee F_{10}F_{11}F_{14(F_{14}=0)} =$$
$$= F_8F_9 \vee F_9F_{11}.$$

Choice of last test point F_{11} gives term F_8F_9. Parallel probing of several terms.

	F_1	F_5	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}
F_1F_{14}	1								1
F_8F_9			1	1					
$F_5F_8F_{14}$		1	1						1
$F_8F_{10}F_{14}$			1		1				1
$F_8F_{12}F_{14}$			1				1		1
F_9F_{11}				1		1			
$F_5F_{11}F_{14}$		1				1			1
$F_{10}F_{11}F_{14}$					1	1			1
$F_{11}F_{12}F_{14}$						1	1		1
F_9F_{13}				1				1	
$F_{13}F_{14}$								1	1
F_1F_9	1			1					

In the instance test points $F_1, F_5, F_{10}, F_{12}, F_{13}$ can be chosen simultaneously. Consequently the covering table will take on form:

	F_8	F_9	F_{11}
F_8F_9	1	1	
F_9F_{11}		1	1

.

Choice of last test point F_{11} gives term F_8F_9 of simplified DNF that confirms the existence the multiple fault in a circuit.

The table below contains information about quantity of elements and terms for a few experiments and a number of necessary probing steps.

Table 1. The Fault DNF Characteristic

Experi-ment #	Experiment quantity	Term quantity	A number of probing	
			Successive	Parallel
1	10	5	4	2
2	9	12	7	2
3	12	9	8	8
4	6	4	2	2
5	10	6	5	3

Effectiveness comparison of various diagnosis methods is shown in Fig. 9 (test points quantity versus Quinn estimation value of a circuit).

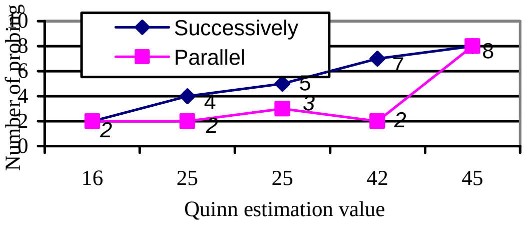


Fig.9. Test points quantity versus Q

As follows from the diagram parallel use of the interactive probing method in several points results in less step quantity of fault DNF analysis procedure. So, it is necessary less time for diagnosis and less costs for SoC design.

A novel matrix diagnosis method for SoC functionality is proposed. It is characterized by use of disjunctive normal form and the fault covering table that enables to obtain full and minimal combinations of multiple faults on basis of use the multiprobing procedures. Proposed successive and parallel methods for testing simulation of SoC are realized in the software. The methods are oriented on embedded Functional Intellectual Property by presented IP-modules.

5 Conclusion

Scientific novelty is the follow. SoC memory in the future will occupy more than 90% of chip area that is oriented on use flexible software. Development of models and methods of quick and exact diagnosis, as well as technologies for repair of faulty cells by on-chip facility in real time and on all life cycle stages of a product are urgent problems. It will enable to decrease quantity of chip pins, to raise yield, to decrease time-to-market, to reduce service costs, as well as to remove output diagnosis and repair facility.

The algebra-logical memory repair method is based on solving of the faulty cells covering problem by spares by means of the Boolean algebra apparatus. The method has quadratic computational complexity and can have hardware or software realization that is service module of fault correction, which enables to carry out memory elements repair in the process of operation.

The classical covering problem use two one-dimensional vectors (X, F), where the covering operator P enables to find minimal subset of the components X, which cover all elements from F: $X_{min} = P(X,F) \leftarrow X \cap F = X_{min}$ by its aggregate functionality. The statement of covering problem of one-dimensional vector F features by two-dimensional matrix $M = (C \times R)$ is needed in reduction of both components to a single metrics (such coordinate system that is common denominator for both structures). Such metrics for the matrix $M = (C \times R)$ and the vector F is one-dimensional structure. So, in this case it is necessary to carry out transformation of two-dimensional structure (memory fault matrix) $M = (C \times R)$ to one-dimensional one by means of the concatenation operation $X = (C * R)$ for subsequent solving of the classical covering problem by application of formal actions, which are defined by the operator $X_{min} = P(X,F)$.

Proposed method of optimal memory fault repair differs from analogs by application of algebra-logical technology of fault covering by two-dimensional memory matrix topology that enables to obtain minimal and full solutions for subsequent repair in real time, which is based on utilization of spares in the form of memory rows and columns.

Practical importance of the research consists of implementation of the method to SoC Functional Intellectual Property Infrastructure. It enables to raise yield essentially (5-10%) on the electronic technology market by means of faulty chip repair in the process of production and operation, as well as to increase the life cycle duration of memory matrixes by repair of them in real time.

On-chip repair is oriented on all objects, which have an address: memory, multiplexers, matrix processors. If it is necessary to repair other structures, they must be designed with an allowance for component addressability. The addressability and regularity of components turns a system into reliable, robust, repairable and durable one.

Further research is oriented on development of testability structure of the system and hardware BIRA module for embedded memory repair in appearance of faults on production and operating stages.

Yervant Zorian (EWDTS' 2007, Yerevan): "Now the main problem of system on a chip repair is development of embedded technologies and methods of the logic repair that occupies no more than 10% of chip area".

References:

[1] J. Bergeron *Writing testbenches: functional verification of HDL models*, Springer, 2003, 512 p.

[2] P. Rashinkar, P. Paterson, L. Singh *System-on-chip Verification: Methodology and Techniques*, Kluwer Academic Publishers, 2002, 393 p.

[3] *IEEE-1800 IEEE Standard for System Verilog Language*, 2005, 586 p.

[4] S. Hamdioui, G. N. Gaydadjiev, A. J. Van de Goor. The State-of-the-art and Future Trends in Testing Embedded Memories, *Records IEEE International Workshop on Memory Technology, Design and Testing*, San Jose, CA, August 2004, pp. 54-59.

[5] Y. Zorian Today's SoC Test Challenges, *ITC International Test Conference*, 2005.

[6] S.Shoukourian, V. Vardanian, Y.Zorian SoC Yield Optimization via an Embedded-Memory Test and Repair Infrastructure, *IEEE Design and Test of Computers*, 2004, pp. 200-207.

[7] L. Youngs, S. Paramanandam Mapping and Repairing Embedded-Memory Defects, *IEEE Design and Test of Computers*, 1997, pp. 18-24.

[8] Y.Zorian, S. Shoukourian Embedded-Memory Test and Repair: Infrastructure IP for SoC Yield, *IEEE Design and Test of Computers*, 2003, pp.58-66.

[9] Y. Zorian, A. Yessayan IEEE 1500 Utilization in SoC Design and Test, *ITC International Test Conference*, 2005.

[10] K. Rossen *Discrete Mathematics and its Applications*, McGraw Hill, 2003, 824 p.

[11] A.N. Parfentiy, V.I. Hahanov, E.I. Litvinova SOC Infrastructure Intellectual Property Models, *ASU and automation devices*, No. 138, 2007, C.83-99.

[12] V.I. Hahanov, I.V. Hahanova *VHDL + Verilog = Synthesis for minutes*, Kharkov: SMIT, 2007, 264 p.

[13] Z. Yervant What is Infrustructure IP?, *IEEE Design & Test of Computers*, May-June 2002, pp. 5-7.

[14] Z. Yervant, G. Dmytris Gest editors' introduction: Design for Yield and reliability, *IEEE Design & Test of Computers*, May-June 2004, pp. 177-182.

[15] IEEE 1500 Web Site. http://grouper.ieee.org/groups/1500/.

[16] D. Densmore, R. Passerone, A. Sangiovanni-Vincentelli A Platform-Based taxonomy for ESL design, *Design&Test of Computers*, September-October 2006, pp. 359-373.

[17] F. DaSilva, Y. Zorian, L. Whetsel, K. Arabi, R. Kapur Overview of the IEEE P1500 Standard, *ITC International Test Conference*, 2003, pp. 988–997.

[18] M.F. Bondarenko, G.F. Krivoula, V.G. Ryabtsev, S.A. Fradkov, V.I. Hahanov *Design and diagnosis of computer systems and networks*, Kiev: NMTS VO, 2000, 306 p.

[19] J.C. Rau, W.T. Huang and C.L. Chien. The optimal testrail architecture for core-based SoC testing, *WSEAS Transactions on Circuits and Systems*, 3(3):720-722, 2004.

[20] Woonchul Ham and Hyunsok Choi. Development of 3D Stereoscopic Camera Interface Based on Embedded System, *WSEAS Transactions on Systems*, 2(6):328-334, 2007.

[21] M. Otesteanu. Embedded Systems with Adaptive Architecture, *WSEAS Transactions on Electronics*, 3(2):100 – 107, 2005.