

コンパクトテスト法を用いたメモリの組込み検査法

広島大学 総合科学部 樹下 行三

1. はしがき

メモリチップの検査については多くの方法が提案されており、対象とする故障に関連して Galloping 法, Walking 法, Marching 法, Walking 法などが知られている [1]~[6]。これらの検査時間は、メモリのビット数を N とすると、 $N\%$, N^2 のオーダーとなる。メモリの記憶容量が増大するにつれて、この検査時間が長くなり、一つのメモリチップが試験機を占有する時間が長くなり、検査費用が増大するという真に重要な問題となりつつある。これを避けるための方法として、検査機構をメモリの中に組み込む方法、即ち、組込み検査法が考えられる [7]。

一方、論理回路を簡単に検査する機構として、コンパクトテスト法が知られている [8]。これは、被検査回路の出力を適当な関数で圧縮し、その値を基準値と比較して、回路の正否を判定しようとするものである。これは、組合せ回路の検査には有効であるが [8]~[13], 検査系列生成自身を別の方法で行なわなければならないこと、また多出力の場合の扱いが効果的でないなどの問題がある。

これに対して、メモリはその内部機構が比較的単純であるので、コンパクトテストの手法を用いるのに適している。検査用の入力系列も、論理回路の場合と違って、規則性の強い系列を用いることが出来、単なる圧縮関数でその正否を容易に判定することが出来る。検査系列生成部と圧縮関数及び判定機構をメモリ内に組み込むことにより、組込み検査を持つメモリを構成することが出来る。

本論文では、 $N = n \times n$ ビットのセルマトリックスメモリチップを対象とする。これを簡単にメモリ書く。また、故障としては、セルの縮退故障、デコーダ故障、隣接パターン依存故障 [14]~[19] を対象とする。

2. セル故障とデコーダ故障

定義 1: メモリにおけるセル故障とは、メモリセルの縮退故障をいう。即ちメモリセルの値が、他のセルに関係なく、固定的に 0 又は 1 となる故障である。

定義 2: メモリにおけるデコーダ故障とは、メモリデコーダにおけるアドレスの誤置状態をいう。アドレスの誤置状態とは、

- 1) 一つのアドレス指定により、 \Rightarrow 以上のアドレスが選択されること。
- 2) 異なったアドレス指定で、一つのアドレスが選択されること。
- 3) どのアドレスも選択されたりすることの起つてくることをいう。

指定アドレスと選択アドレスの関係を図1のように表わし、これをアドレス対応 Γ とする。即ち、この対応が 1 対 1 でないとき、デコーダ故障となっている。

多重選択が行なわれたとき、書き込みには、選択されたすべてのアドレスにデータが書き込まれるものとする。読み出しの時には、選択され

複数のアドレスの内容の論理積となるか論理和となるかのいずれかでありと仮定する。前者をAND型、後者をOR型読出しというとする。

メモリのアドレスを j (又は (i, j)) とするとき、 j ((i, j)) 番地に z を書き込む命令を $W_z(j)$ ($W_z(i, j)$) で、その読み出し命令を $R(j)$ ($R(i, j)$) で表わすことにする。

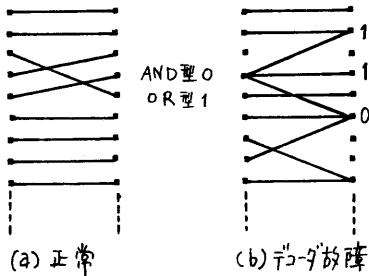


図1 アドレス対応グラフ

検査系列は、書き込み命令と読み出し命令より成ると考へる。書き込み命令の系列が検査入力系列、読み出し命令の系列が検査出力系列に対応する。ゴニパフテストの手法を適用し、次のように検査可能性を定義する[8]。

定義3: メモリの検査系列を T 、その出力系列を R とする。圧縮関数 C を用いて、 $C(R)$ の値と基準値が異なるときは限り故障が存在すると判定するこゝが出来るとき、メモリは検査系列 T と圧縮関数 C により検査可能であると行う。

定義4: 系列 R に対して圧縮関数 C を次のように定義する。

$$C_1(R) = \sum_{j=0}^n r_j, \quad R = \prod_{i=0}^n r_i = r_0 r_1 \dots r_n$$

補題1: メモリのセル故障は、圧縮関数 C_1 と検査系列 T_C により検査可能である。 $\equiv \equiv$ で、

$$T_0 = W_0 R_0 W_1 R_1 \dots$$

$$W_z = \prod_{j=0}^{N-1} W_z(j), \quad (z=0,1)$$

$$R_0 = \prod_{j=0}^{N-1} R(j), \quad R_1 = \prod_{j=0}^{N-1} R(j)$$

で、基準値は、 $C_1(R_0) = C_1(R_1) = 0$ である。証明) 定義より明らかである。

定義5: 次の系列を定義する。

$$U_i = W_0(0) \dots W_0(i-1) W_1(i) W_0(i+1) \dots W_0(N-1)$$

$$R_N = \prod_{j=0}^{N-1} R(j)$$

補題2: メモリのデコ-ダ故障は圧縮関数 C と検査系列 T_d により検査可能である。 $\equiv \equiv$ で、

$$T_d = \prod_{i=0}^{N-1} (U_i R_N)$$

であり、基準値はオベアの i につき、 $C_1(R_N) = 1$ となることである。

証明) デコ-ダ故障で、アドレスの無選択が起るときにはその読出しにより故障の存在を知ることが出来る。無選択が存在しないとするとき、図2に示す形の対応をもつアドレスグラフが存在する。

まずAND型の場合に於いて考へる。この時には、対応線 "a" が最も若い番地にあるものについて考へる。図2に示す系列 U_j を書き込み、それを読出すと、オベアが0であ

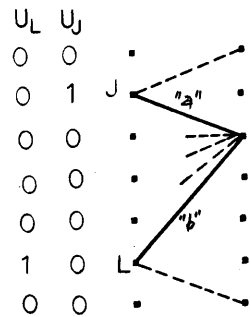


図2 デコ-ダ故障の検出

る系列となり $C_1(R_N)$ は 1 ではない。

OR 型の場合には、対応線 "b" が最も大きい番地になる場合を考へる。図 2 に示す系列 U_i を書き込み、それを読み出すと、1 の数が 2 以上となり、 $C_1(R_N)$ の値は 1 でなくなる。

いざこれにしても、 U_i に対する読み出しの圧縮関数値は基準値と異なり、デコード故障は検出可能である。 (証明終)

検査系列 T_d は、その中に各セルについて 0, 1 の書き込み、読み出しを含まない。補題は、次の定理に拡張される。

定理 1: メモリのセル故障とデコード故障は、圧縮関数 C_1 と検査系列 T_d により検査可能である。基準値は、すべての i につき、 $C_1(R_N)$ が 1 となることである。

(証明) 補題 2 より、セル故障とデコード故障が共存する場合について証明すればよい。

この場合には、図 3 に示すような検出不可能な部分が存在する。検出不可能部の右側が他と異なれば (矢線部分) これは検出可能なものになる。検出不可能部は、右側の数が左側より 1 つ多いので、不可能部を除いたアドレス対応グラフは、無選択を含むが、図 2 の形を含むかのいずれかになる。図の場合にあって、その中に縮退故障が存在しても検査可能であることは容易に示される。 (証明終)

定義 6: 検査系列に含まれる読み出し命令と書き込み命令の数を検査系列長と定義する。

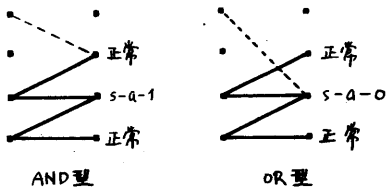


図 3 検出可能故障

系: T_d の検査系列長は $2 \cdot N^2$ である。

検査系列 T_d の生成部と圧縮関数 C_1 を用いた判定部に相当する付加論理回路は、図 4 に示すようになり、 $\log_2 N$ ビットカウンタが 3 個、7 1170 フロップ 2 個程度の付加で実現出来る。メモリのビット数 N に対して、付加的な記憶ビット数は、約 $3 \log_2 N$ であるから付加量の割合は、 $3 \log_2 N / N$ であるから、 N が大きくなれば十分に引き合えるものと考へられる。例えば、64K-bits で、付加量は 0.07% である。

検査系列長が $2 \cdot N^2$ であることより、検査実行時間はかなり長くなる。しかしこれは、組込み検査としてメモリ 4 μ 7 内に内蔵されておるので、特別の試験機を必要とせず、各 4 μ 7 を同時に検査する事が出来、検査時間は、単に製造と出荷の間の遅延時間という事になる。したがって、数時間の検査という事は全く不可能ではないであろう。実用的には、バーインテストなどと組合せればより効果的であろう。

しかし、メモリの読み書きのサイクル時間を 100 ns とすれば、 N が 1M bits で、約 60 時間、16M bits になれば約 1 年となり、組込み検査としても実用性がなくなる。

次に、更に検査系列長を短くする方法について考へよう。

定義 7: $0(1)$ が n 個並んだ系列を $0^n(1^n)$ と書く。

$$x_k = (0^{2^k} 1^{2^k})^{N/(2^k+1)}, y_k = (1^{2^k} 0^{2^k})^{N/(2^k+1)}$$

とする。ここで、 N は 2^p の数であり、 k は $0, 1, \dots, \log_2 N - 1$ の値とする。系列 $X_k (Y_k)$ をこの順に $0 \sim N-1$ 番地に書き込む系列を $W(X_k) (W(Y_k))$ と表わす。

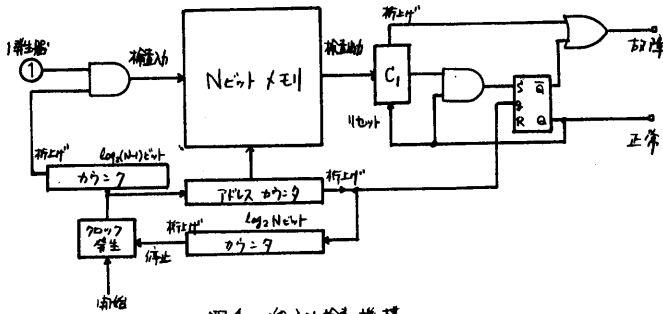


図4 組込検査機構

定理2: メモリのセル故障とデコード故障は、検査系列 T_e によって検査することが出来る。ここで、

$$T_e = \prod_{k=0}^{\log_2 N - 1} (W(X_k) R_N) \prod_{k=0}^{\log_2 N - 1} (W(Y_k) R_N)$$

$$R_N = \prod_{i=0}^{N-1} R(i)$$

である。

証明) セル故障だけの場合は明らかである。まずデコード故障だけの場合を考える。無選択アドレスが存在しないとしたら、図5に示す形のアドレス対称が存在する。

AND型の場合にはJが最も若い番地となるものを考える。ここで、J番地に1、L番地に0を書き込み、これを読出すとJ番地では0が読出される。

OR型の場合

には、Lが最も

大きい番地であ

るものを考え、

J番地に0、L

番地に1を書き

込む。これを読

み出すと、J番

地の内容は1と

なり、最初に書

き込んだ値と異

なる。

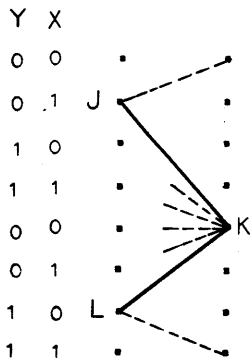


図5 デコード故障の検出

検査系列 T_e の書き込み系列は、任意に異なる二つのアドレスに対して、それぞれ0, 1および1, 0を書き込むようになってくる。

したがって、 T_e によって、任意のデコード故障を輸出することが出来る。

セル故障とデコード故障が共存する場合には、定理

1の証明で示したのと同様に、

検出不可能部を除いて考えると、図5の場合で縮退故障の存在を考慮

することになる。この場合にも上の議論をそのまま適用出来る、書き込みの値と読

み出しの値が異なるものが存在することになる。したがって、入力系列と出力

系列が等しくなりという意味で検査可能となる。

(証明終り)

系: T_e の検査系列長は $4N \log_2 N$ である。

検査入力系列, X_k, Y_k は、規則性のある系列であり、これを簡単に構成することが出来ることが知られており [18] (図6)。例之ば、 $N=8$ の場合には次に示すようになる。

$$X_1 = 01010101$$

$$X_2 = 00110011$$

$$X_3 = 00001111$$

$$Y_1 = 10101010$$

$$Y_2 = 11001100$$

$$Y_3 = 11110000$$

しかし、読み出した系列が系列が二の形であることを判定する簡単な圧縮

関数は知られていない。したがって、この場合には、入力系列と出力系列を

直接比較することによって検査を行う方法を用いることになる。検査系列の

生成部は図6のように容易に実現され

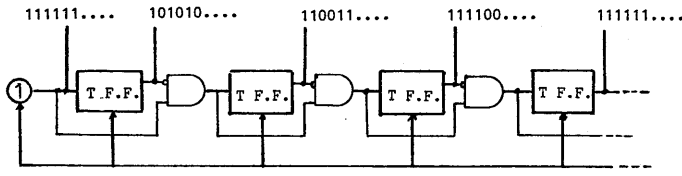


図6 系列 Y_k の発生

るので、組込み検査として実現することは容易である。

コンパクトテストの手法を用いる特長は、圧縮関数による判定が、基準値だけで行なえるので、入力側との同期性はそれほど厳格にならなるとしてある。しかし、入力系列と出力系列を直接比較するときには、同期又は記憶の機構が必要となる。

定理2の方法では、検査系列長は、 $4N \log_2 N$ であるから、かなり大きな N に対しても実現可能である。メモリのサイクル時間も 100 ns とし、 1 Mbits の場合で 8.4 sec 、 16 Mbits で 161 sec 程度であるから、組込み検査としては十分に実用的な値である。

3. 隣接109-依存故障

この節では、隣接109-依存故障に対するコンパクトテスト法を用いた組込み検査の手法について考察する。

定義8: メモリの一つのセル (i, j) に対して、 $(i, j-1)$ 、 $(i, j+1)$ 、 $(i+1, j)$ 、 $(i-1, j)$ なる4つのセルを (i, j) の隣接セルとす。隣接セルに (i, j) を加えて、1隣接セルとす。

定義9: セル (i, j) に書き込みを行うとき、1隣接セルの値の影響で、正しい値が書けないうち、その隣接セルの値が変わるとき、この故障を1隣接109-依存故障とす。これを簡単に1-APS故障とす。

1-APS故障は、セル (i, j) に対して、その1隣接セルが可能なすべての値を取るようになり、セル (i, j) に0を書く場合と1を書く場合を尽せばよい。そして、これをすべてのセルについて繰返せばよい。したがって、次の検査系列

R_M によって1-APS故障は検査される。

補題3: X メモリの1-APS故障は、検査系列 R_M によって検査される。

$$R_M = \prod_{i,j=0}^{n-1} \prod_{z=0}^1 \prod_{x=0}^1 P_w(i, j; x) W_z(i, j) Pr(i, j)$$

ここで、

$$P_w(i, j; x) = W_a(i, j-1) W_b(i-1, j) W_c(i, j+1) W_d(i+1, j) W_e(i, j),$$

$$Pr(i, j) = R(i, j)R(i, j-1)R(i-1, j)R(i, j+1)R(i+1, j)$$

であり、 $abcde$ は X の2進表示である。

系: 検査系列 R_M の長さは $704N$ である。

検査系列 R_M には、生成109-に重複が含まれており、読み出し系列自身も圧縮関数を用いるような規則的なものではない。この点を考慮して R_M を改良して次の R_S を得る。

補題4: X メモリの1-APS故障は、検査系列 R_S により検査される。

$$R_S = \prod_{x=0}^1 \prod_{b=0}^1 [(U_1(AB)U_3(CD)U_2(aa)U_4(aa)U_2(bb)$$

$$U_4(bb) V_0 V_1 V_2 V_3)(U_2(AB)U_4(CD)U_1(aa)U_3(aa)U_1(bb)U_3(bb) V_0 V_1 V_2 V_3)],$$

ここで、 $ABCD$ は X の2進表示であり、 U_j の値は、 $j=0, 1, 2, 3$ に対して次のように決められる。

$$u_j(AB) = u_j(AB) u_{j+4}(AB) \dots u_{j+n-4}(AB),$$

$$u_j(AB) = W_A(0, j) W_B(1, j+1) W_A(2, j+2) \dots \\ \dots W_A(n-2, j+n-2) W_B(n-1, j+n-1)$$

$$v_j = v_j v_{j+4} \dots v_{j+n-3},$$

$$v_j = \prod_{i=0}^{n-1} R(1, i+j).$$

証明) R_S の前半の [] 内の系列は、
図7 のようになったらいい ($n=8$ の場合)。

したがって、
ABCD の各の値
のすべての組
合せに対して、
位置 E に 0 と
1 の値を書き
込めはよい。

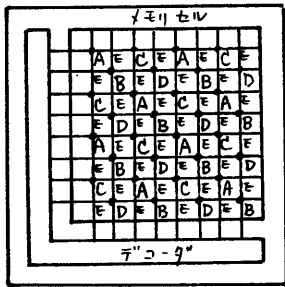


図7 検査系列 R_S

後半の []
は、ABCD
の位置と E の
位置を交代し
たものである。

したがって、 R_S によりすべての位置
に対して 1-APS 故障を検査すること
ができる。 (証明終)

系: 検査系列 R_S の長さ n は $320N$ である。

この検査により読み出される出力系列
は、正常な場合は、 0^{2p} , 1^{2p} , $(01)^p$,
 $(10)^p$ の形である。従って、出力系列
がこの形であるかどうかを識別すること
により、1-APS 故障を検査すること
ができる。

定義10: $L(2p)$ を次の形の系列の集合
とする。

$$L(2p) = [0^{2p}, 1^{2p}, (01)^p, (10)^p].$$

定義11: $L(2p)$ に属す正常出力が $L(2p)$
の他の系列になるような 1-APS 故障を、

$L(2p)$ 故障と 11 号とにする。

補題5: R を長さ $2p$ の 2 進系列で
あるとする。 $R \in L(2p)$ であるための
必要十分条件は、 $C_{23}(R) = 0$ となるこ
とである。 $\equiv \equiv$

$$C_{23}(R) = C_2(R) C_3(R),$$

$$C_2(R) = \sum_{i=2}^{2p} (r_{i-1} \cdot r_i),$$

$$C_3(R) = \sum_{i=2}^{2p} (r_{i-1} \cdot r_i \cdot 1).$$

である。

証明) $C_2(R) = 0$ であるための必要
十分条件は、系列に変化がないこと
あるから、 R は 0^{2p} 又は 1^{2p} である。
 $C_3(R) = 0$ であるための必要十分条件
は、系列の中に同じものが続かないこ
とであるから、 R は $(01)^p$ 又は $(10)^p$
となることである。従って、 $R \in L(2p)$
のための必要十分条件は $C_{23}(R) = 0$
となることである。 (証明終)

定理3: メモリの $L(2p)$ 故障でない
1-APS 故障は、検査系列 R_S と圧縮
関数 C_{23} により検査可能である。 C_{23}
に対する基準値は V_j ($j=0, 1, 2, 3$) につ
いて 0 である。

証明) 補題 4, 5 および定義 10 により
明らかである。 (証明終)

更に、 $L(2p)$ 故障を見つける必要の
あるときには、圧縮関数を追加する必
要がある。

定義12: 次の圧縮関数を定義する。

$$C_{6x}(R) = C_2(xR)$$

$$C_{7x}(R) = C_3(xR),$$

$\equiv \equiv$, x は 0, 1 の値をとる。

補題6: $R \in L(2p)$ の2進系列とするとき, 次の関係が成立つ.

$$R = 0^{2p} \quad \text{if and only if} \quad C_{60}(R)=0,$$

$$R = 1^{2p} \quad \text{if and only if} \quad C_{61}(R)=0,$$

$$R = (01)^p \quad \text{if and only if} \quad C_{70}(R)=0,$$

$$R = (10)^p \quad \text{if and only if} \quad C_{71}(R)=0.$$

証明) $C_{60}(R)$ は R の最初の値が0でその後に変化がない時に限りその値が0となる。 $C_{61}(R)$ は R の最初の値が1でその後に変化がない時に限りその値が0となる。 $C_{7x}(R)$ についても同様に示される。
証明終り

定理4: メモリの1-APS故障は, 検査系列 R_S と圧縮関数 $C_{60}, C_{61}, C_{70}, C_{71}$ により検査可能である。

圧縮関数値の基準値はすべての V_i に対して0である。

証明) 定理3と補題6より明らかである。
証明終り

検査系列 R_S の書き込み系列は, その構成から分子のように, $L(2p)$ の系列の繰返しと成り立っている。したがって, 検査系列生成部は, 簡単な信号発生回路で構成する事が出来る。

圧縮関数は, $L(2p)$ 故障を無視すれば C_{23} , そうでなくても C_{6x}, C_{7x} での11の 2^p の回路は簡単に構成出来る。したがって, 組込み検査として付加になるリードウェアはそれほど多くはない。

4. おまじ

以上, セルマトリックスメモリアップに於いて, セル故障, テーコーダ故障, 隣接パターン依存故障に対して, それを検査する方法および組込み検査メモリとして実現する方法について考察し, 比較的簡単に構成出来ることを示した。

本論文では, それぞれの故障に対する組込み検査について示したが, それを総合的に実現することはそれほど困難ではないであろう。実際の実現に当っては, 検査回路についての耐故障性についても考慮する必要がある。

更に多くの固定故障に適用出来るように拡張することは, より簡単な検査機構で, より短い検査系列を求めることが問題として残されてくる。

終りに当って, 本考察について御計論いただいた, 大阪大学工学部藤原秀雄博士に感謝する。

文 献

- [1] W.G. Fee(Ed.): LSI Testing (Tutorial); Section 3, "Memory Testing," IEEE Catalog No.EHO 122-2, 1977.
- [2] W. Barraclough, A.C.L. Chiang and W. Sohl: "Techniques for Testing the Microprocessor Family," Proc. IEEE, Vol.64, No.6, June 1976.
- [3] J.M. Crafts: "Technique for Memory Testing," IEEE Computer, Vol.12, No.10, October 1979.
- [4] S.M. Thatte and J.A. Abraham: "Testing Semiconductor Random Access Memories," Proc. FTCS-7, 1977.
- [5] R. Nair, S.M. Thatte and J.A. Abraham: "Efficient Algorithm for Testing Semiconductor Random Access Memories," IEEE Trans. on Computers, Vol.C-27, No.6, June 1978.
- [6] 岸政七: "Waltzing Patternを用いたICメモリ素子試験" 電子通信学会論文誌, Vol.J60-D, No.12, Dec. 1977.

- [7] V.K. Agarwal and E. Cerny: "Store and Generate Built-in-Testing Approach," Proc. FTCS-11, June 1981.
- [8] H. Fujiwara and K. Kinoshita: "Testing Logic Circuits with Compressed Data," J. of Design Automation and Fault-Tolerant Computing, Vol.3, No.3-4, Winter 1979.
- [9] J.P. Hayes: "Transition Count Testing of Combinational Logic Circuits," IEEE Trans. on Computers, Vol.C-25, No.6, June 1976.
- [10] S.C. Seth: "Data Compression Techniques in Logic Testing: An Extension of Transition Counts," J. of Design Automation and Fault-Tolerant Computing, Vol.1, No.2, February 1977.
- [11] S.M. Reddy: "A Note on Testing Logic Circuits by Transition Counting," IEEE Trans. on Computers, Vol.C-26, No.3, March 1977.
- [12] J.P. Hayes: "Generation of Optimal Transition Count Tests," IEEE Trans. on Computers, Vol.C-27, No.1, January 1978.
- [13] K.P. Parker: "Compact Testing: Testing with Compressed Data," Proc. FTCS-6, June 1976.
- [14] J.P. Hayes: "Detection of Pattern Sensitive Faults in Random-Access Memories," IEEE Trans. on Computers, Vol.C-25, No.10, October 1975.
- [15] J.P. Hayes: "Testing Memories for Single Cell Pattern-Sensitive Faults," IEEE Trans. on Computers, Vol.C-29, No.3, March 1980.
- [16] D.S. Suk and S.M. Reddy: "Test Procedure for a Class of Pattern-Sensitive Faults in Semiconductor Random-Access Memories," IEEE Trans. on Computers, Vol.C-29, No.6, June 1980.
- [17] V.P. Srin: "APL Test for RAM chips," IEEE Computer, Vol.10, No.7, July 1977.
- [18] 村上, 梅村, 尾崎: "系列発生器の構成に関する考察" 電子通信学会論文誌, Vol.C-53, No.1, Jan. 1970.