

# 一种新型的仿生电子细胞基因存储结构

蔡金燕, 朱 赛, 孟亚峰

(机械工程学院电子与光学工程系, 河北石家庄 050003)

**摘 要:** 基因存储是电子细胞的重要组成部分, 已有的基因存储无法兼顾系统的可靠性和硬件消耗. 设计了一种新型的基因存储结构, 细胞采用相关冗余方式存储系统的部分基因. 通过基因更新过程, 基于相邻细胞的基因信息恢复故障细胞损失的基因. 细胞内存储基因数目与阵列和目标电路规模无关, 可由设计者根据系统需求设置. 理论分析和仿真实验表明, 该基因存储不仅实现了阵列功能分化和自修复, 而且可在保持系统可靠性的前提下, 降低基因存储的硬件消耗, 可用于大规模仿生自修复芯片的设计.

**关键词:** 胚胎电子阵列; 自修复; 部分基因循环存储; 基因移位; 基因备份数目

**中图分类号:** TP302.8 **文献标识码:** A **文章编号:** 0372-2112 (2016)08-1915-09

**电子学报 URL:** <http://www.ejournal.org.cn> **DOI:** 10.3969/j.issn.0372-2112.2016.08.021

## A Novel Gene Memory Structure for Bio-inspired Electronic Cell

CAI Jin-yan, ZHU Sai, MENG Ya-feng

(Department of Electronic and Optical Engineering, Ordnance Engineering College, Shijiazhuang, Hebei 050003, China)

**Abstract:** Gene memory is an important part of electronic cell, and the existing genes memory structure cannot take into account the reliability and hardware cost. A novel gene memory structure named partial-DNA cyclic memory was presented, and only part of the system's entire DNA was stored with correlated redundancy. The lost genes caused by faulty cells could be obtained through gene update according to the faulty cells' neighbor cells. The stored gene number is independent from the scale of embryonics array and target circuit, and can be set according to actual demand in design process. Theoretical analysis and simulation results show that the novel gene memory not only to achieve the embryonics array's functional differentiation and self-repair, but also reducing hardware overhead while keeping system reliability. The proposed gene memory can be used to design larger scale self-repair chip.

**Key words:** embryonics array; self-repair; partial-DNA cyclic memory; gene shift; gene backup number

### 1 引言

胚胎电子阵列是一种新型的仿生电子硬件, 基于该硬件实现的目标电路具有快速自修复能力<sup>[1]</sup>. 胚胎电子阵列中基本单位为仿生电子细胞, 经典的仿生电子细胞中, 95%的硬件消耗来源于基因存储<sup>[2]</sup>, 研究者以降低硬件消耗为目的, 在全存储基础上, 提出了多种基因存储结构, 如只存储所在行/列表达基因的行/列存储与循环变形存储<sup>[3]</sup>、只存储部分相邻细胞表达基因的部分基因存储<sup>[4]</sup>、只存储自身和其后一个邻居细胞的循环备份存储<sup>[5]</sup>、原核存储<sup>[6,7]</sup>等.

全存储、行/列存储、循环变形存储及部分基因存储中采用硬件冗余方式, 硬件消耗大. 同时, 用于通用自修

复芯片设计时, 芯片内存储规模与芯片规模呈非线性增长关系, 当芯片规模较大时, 所需基因存储规模巨大. 循环备份存储、原核存储中采用相关冗余方式, 降低了硬件消耗, 但由于基因备份数目较少, 系统可靠性较低.

水平基因转移 (Horizontal Gene Transfer, HGT) 能跨越物种隔离, 在亲缘关系或远或近的生物有机体间进行遗传信息转移. 水平基因转移能帮助受体生物绕过通过点突变和重组创造新基因的缓慢过程, 从而加速基因组的革新和进化<sup>[8]</sup>. 对于生物菌群, 通过高效的基因水平转移, 遗传物质频繁交换, 其致病性和耐药性不断提高<sup>[9]</sup>.

早期真核生物以单细胞形式存在, 该阶段的另一种形式的基因转移—胞内基因转移 (Intracellular Gene

Transfer, IGT)对整个真核生物类群的进化产生了深远影响<sup>[8]</sup>.

受 HTG 和 IGT 现象的启发,提出了一种新型的基因存储结构—部分基因循环存储,该存储采用相关冗余方式进行基因存储,通过细胞的 IGT、细胞间的 HTG 进行基因操作,在自修复过程中根据周围细胞进行细胞存储的更新,且基因备份数目可以根据实际需求设置,在保持系统可靠性的同时,降低了基因存储的硬件消耗.电子细胞只存储自身和其后若干相邻细胞的表达基因,所存储的基因数目与阵列规模无关,可用于大规模通用自修复芯片的设计.

### 2 部分基因循环存储结构

#### 2.1 细胞及基因存储结构

部分基因循环存储中,细胞内存储自身和其后连续  $k-1$  个细胞的表达基因, $k$  为基因备份数目,根据目标可靠性及硬件消耗需求在设计阶段确定.对于具有  $n$  个细胞的阵列,第  $i$  个细胞的表达基因记为  $g_i, 1 \leq i \leq n$ ,则细胞  $C_i$  内的存储基因为  $g_i \sim g_{i+k-1}$ .

基于部分基因循环存储的细胞结构如图 1 所示.由逻辑模块(Logical Block, LB)、开关盒(Switch Box, SB)、内建自测试(Built-In Self-Test, BIST)、基因存储单元(Memory Unit)和状态控制电路(State Control Circuit)组成. LB、SB 和 BIST 采用已有研究成果<sup>[10]</sup>,本文不再赘述.

图 1 中所示  $C_i$  细胞结构中,RunState 为阵列运行状态信号,当 RunState = 0 时,阵列为功能分化状态,RunState 为 1 时,阵列处于运行状态; $C_{i-1}SE$  为细胞  $C_{i-1}$  的移位使能信号, $C_{i-1}SE = 1$ ,细胞  $C_{i-1}$  的存储单元可以移位,否则,其存储单元停止移位; $C_{i+1}FS, C_{i+2}FS, \dots, C_{i+k-1}FS$  为其后相邻的  $k-1$  个细胞的故障信号, $C_jFS = 1$  表示细胞  $C_j$  发生故障,否则, $C_j$  正常.

基因存储单元存储细胞所需基因,GI 为输入基因,

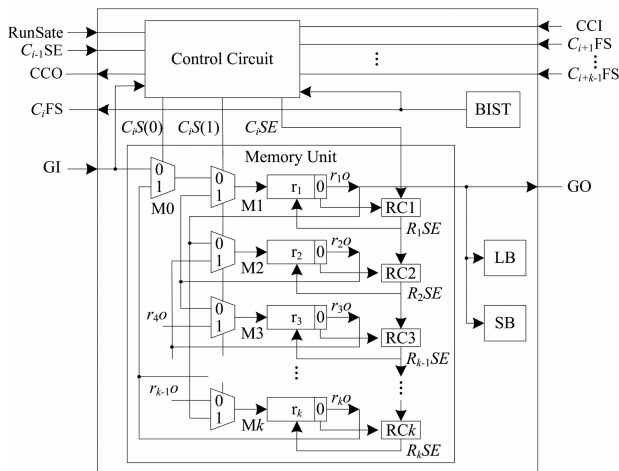


图1 具有部分基因循环存储的细胞结构

GO 为输出基因;基因存储单元具有  $k$  个基因寄存器  $r_1, r_2, \dots, r_k$ ,每个寄存器存储一个基因,且  $r_1$  存储细胞的表达基因;第  $i$  个寄存器  $r_i$  的输出为  $r_i o$ ;寄存器的 0 位为基因标记, $r_i o(0) = 0$  时  $r_i$  内为空闲基因, $r_i o(0) = 1$  时  $r_i$  内为功能基因; $r_i$  的输入由 2-1 多路复用器  $M_i$  选择,以控制存储单元中基因的移动方向.每个寄存器移位与否可单独控制,其移位使能信号  $R_iSE$  由第  $i$  个寄存器控制电路(Register Control Circuit, RC)  $RC_i$  产生, $R_iSE = 1$  时  $r_i$  可以移位,否则  $r_i$  不能移位.

状态控制电路根据阵列状态和其后  $k-1$  个后续细胞的状态产生存储单元的移位使能信号和移位控制信号,控制存储单元内基因的移位,完成阵列的功能分化和自修复.

#### 2.2 阵列中的基因存储串

阵列中相邻细胞的 GI 端和 GO 端相连,组成图 2 所示环状结构,所有细胞内的基因存储构成基因存储串.

阵列具有一个母细胞,进行与外界信息交互,如图 2 中(1,1)细胞.母细胞通过 2-1 多路复用器  $M$  根据阵列运行状态信号 RunState 选择输入.当阵列为功能分化状态,母细胞的输入为基因配置串 ConfGene;阵列处于运行状态时其输入为基因存储串末端细胞的输出,整个阵列内的基因存储构成环状,进行基因的传递.

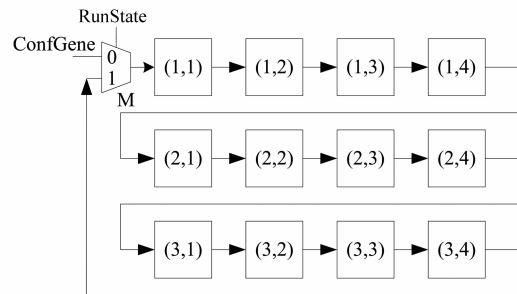


图2 阵列中的基因存储串

#### 2.3 基因移位

与 IGT 和 HTG 类似,阵列的分化及自修复通过细胞内和细胞间基因移位完成.

细胞内基因移位包括顺时针非循环移位、顺时针循环移位和逆时针循环移位,在移位控制信号  $C_iS$  的控制下执行.  $C_iS = 00$  时,细胞进行顺时针非循环移位,在时钟驱动下, $r_1$  内基因移至  $r_2$ 、 $r_2$  内基因移位至  $r_3$ 、 $\dots$ 、 $r_{k-1}$  内基因移位至  $r_k$ , $r_k$  上基因被覆盖,如图 3(a) 所示;  $C_iS = 01$  时执行顺时针循环移位,在顺时针非循环移位的基础上, $r_k$  上的基因移位到  $r_1$ ,如图 3(b) 所示;当  $C_iS = 1x$  时,为逆时针循环移位模式, $r_k$  内基因移至  $r_{k-1}$ 、 $r_{k-1}$  内基因移至  $r_{k-2}$ 、 $\dots$ 、 $r_2$  内基因移位至  $r_1$ 、 $r_1$  内基因移位至  $r_k$ ,如图 3(c) 所示.

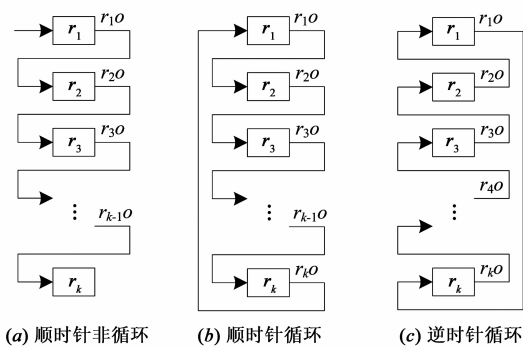


图3 基因移位方式

顺时针非循环移位中,通过控制寄存器的移位使能信号,可以使部分寄存器参与移位。

细胞间基因的移位通过图2所示基因存储串进行,通过GI接收前一细胞的输入,并由GO将基因信息传递至后一细胞。

### 3 控制电路

控制电路产生存储单元的移位控制信号及寄存器移位使能信号,完成阵列的功能分化和自修复过程,包括状态控制电路和寄存器控制电路。

#### 3.1 阵列的分化和自修复

##### 3.1.1 阵列的分化过程

对于具有  $n$  个电子细胞、基因备份数目为  $k$  的阵列,分化所需基因配置串为  $g_1 g_2 \cdots g_n g_1 g_2 \cdots g_{k-1}$ 。

在阵列全局时钟的控制下,基因配置串以  $g_{k-1}$  到  $g_1$  的次序送入母细胞,通过细胞间和细胞内的移位将各细胞的表达基因移位至其  $r_1$  寄存器,完成阵列的分化:细胞间基因由母细胞的GI开始,依次在  $C_1, C_2, \dots, C_n$  等细胞的  $r_1$  寄存器上扩散;同时阵列中所有细胞执行顺时针非循环移位,基因信息在细胞内的  $r_1, r_2, \dots, r_k$  寄存器上传递。经过  $n+k$  个周期的移位,所有基因移位到相应的寄存器中,阵列分化过程完成。

##### 3.1.2 阵列的自修复过程

阵列的自修复是在控制细胞的控制下,通过细胞基因存储的更新完成的。阵列中的每个细胞均具有控制能力,自修复过程中控制细胞及需要更新基因的细胞更新范围确定规则如下:

**规则1:**当阵列中出现一个故障细胞时,位于故障细胞前的正常细胞为控制细胞,更新范围包括位于故障细胞后的正常细胞及一个空闲细胞;

**规则2:**当阵列中出现多个相邻细胞故障时,位于故障细胞前的正常细胞为控制细胞,更新范围为位于故障细胞后的正常细胞及与故障细胞相同数目的空闲细胞;

**规则3:**当阵列中出现两个不相邻的细胞同时发生

故障时,首先按照规则1对位置较为靠后的故障细胞进行操作,操作完成后,再按照规则1对位置靠前的故障细胞进行操作。

在控制细胞的控制下,阵列开始自修复过程:首先控制细胞计算其后  $k-1$  个细胞中故障细胞数目,记为  $fcn, fcn \leq k-1$ ,自身基因进行  $fcn$  次逆时针循环移位;其次控制自身及更新范围内的细胞进行基因更新,同时进行  $fcn$  次移位,控制细胞进行顺时针循环移位,其他细胞进行顺时针非循环移位。

基因更新过程中,故障细胞处于“透明”状态,其基因存储不参与移位,控制细胞内的基因直接传递到故障后面的细胞;基因更新过程中,更新范围内细胞中空闲基因后的寄存器不参与移位。

通过以上操作,可以完成阵列的自修复,耗时  $(fcn + fcn = 2 * fcn)$  个时钟周期。

#### 3.2 状态控制电路

对于图1所示  $C_i$  细胞,CCI为后方细胞控制角色信号,CCI=1表示后方存在控制细胞,否则,后方细胞全部为非控制细胞;CCO为输出控制角色信号,该信号为前方细胞的CCI信号。设置控制角色信号CE,CE=1则  $C_i$  为控制细胞,否则  $C_i$  不是控制细胞。

##### 3.2.1 电子细胞状态

由3.1节可知,基于本文基因存储的电子细胞在运行过程中具有功能分化、正常运行及自修复三种状态。

阵列进行3.1.1节所述的功能分化时,所有细胞处于功能分化状态。基因配置信息由母细胞开始在基因配置串上的细胞及细胞内部的寄存器间扩散,所有细胞的存储单元执行顺时针非循环移位操作;

阵列无故障运行过程中,所有细胞处于正常运行状态,执行3.1.2所述控制细胞判断状态,细胞存储单元无操作;

阵列中细胞发生故障时,控制细胞及更新范围内的细胞转至自修复状态,状态控制电路产生控制信号完成修复过程所需基因移位操作。

##### 3.2.2 状态控制电路功能与实现

为了完成细胞整个生命周期的控制,状态控制电路需要具有控制角色判断、存储单元控制功能。

正常运行状态下,状态控制电路根据控制角色判断规则进行控制角色信号的生成,由3.2.1节可知,角色判断功能为

$$CE = \overline{C_i FS} * C_{i+1} FS * \overline{CCI} \quad (1)$$

功能分化、自修复状态下,进行各种基因操作,的状态控制电路在阵列运行状态信号RunState、细胞控制角色信号CE、前面细胞移位使能信号  $C_{i-1} SE$  及其状态信号  $C_{i-1} ST$  决定存储单元的移位使能信号  $C_i SE$ 。具体规则如表1所示。

表 1 存储单元移位使能信号确定规则

输入信号		输出信号		
RunState	CE	$C_{i-1}SE$	$C_{i-1}ST$	$C_iSE$
0	x	x	x	1
1	1	x	x	0
1	0	0	x	0
1	0	1	1	1
1	0	1	0	0

细胞状态信号  $C_{i-1}ST = 0$  时,细胞处于空闲状态,其表达基因为空闲基因,  $C_{i-1}ST = 1$  时细胞处于激活状态,其表达基因为功能基因,因此细胞的状态信号与其表达基因的标识一致,有  $C_{i-1}ST = C_{i-1}r_1o(0)$ . 则状态控制电路中移位使能信号产生电路功能为

$$C_iSE = \overline{RunState} + CE + C_{i-1}SE * C_{i-1}r_1o(0) \quad (2)$$

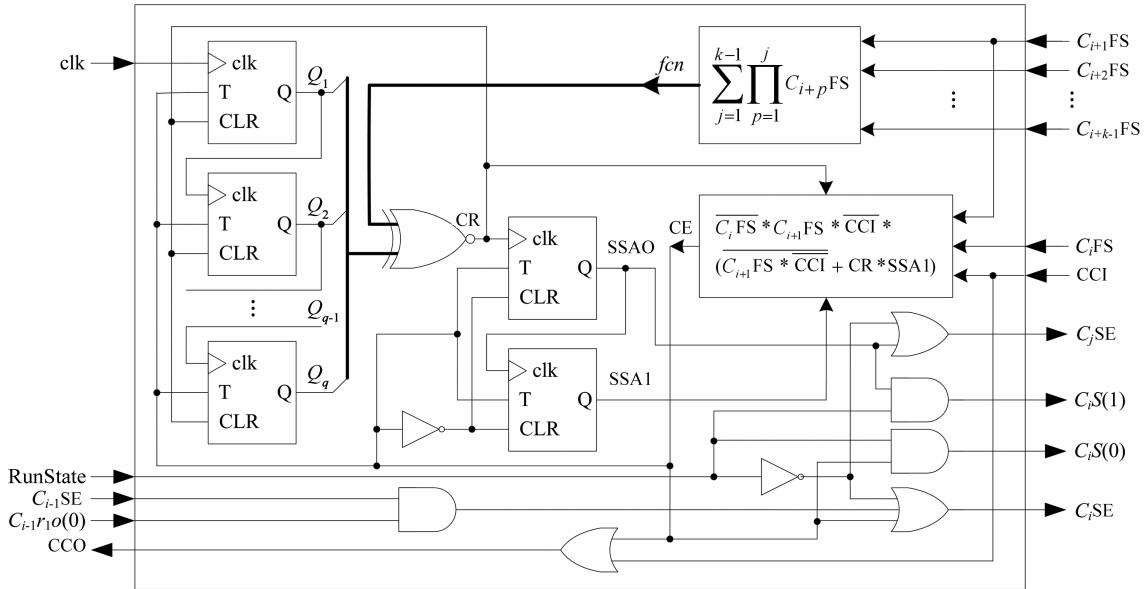


图4 修复控制电路

自修复过程中,控制细胞、更新范围内细胞进行基因移位时,移位次数与故障细胞数目相关,故障细胞数目  $fcn$  计算为

$$\begin{aligned} fcn &= C_{i+1}FS + C_{i+1}FS * C_{i+2}FS + \dots \\ &+ C_{i+1}FS * C_{i+2}FS * \dots * C_{i+k-1}FS \\ &= \sum_{j=1}^{k-1} \prod_{p=1}^j C_{i+p}FS \end{aligned} \quad (3)$$

移位过程中,使用计数器进行移位次数的计数.在更新准备阶段,到移位次数达到  $fcn$  时,停止移位,开始基因更新阶段,并重新开始计数,当再次计数到  $fcn$  时,自修复完成.

为了避免同时出现两个控制细胞,阵列中一旦出现控制细胞,便将控制细胞信号向前传递,由下式实现

$$CCO = CCI + CE \quad (4)$$

实现上述功能的状态控制电路如图 4 所示.

### 3.3 寄存器控制电路

在功能分化及自修复过程中的控制细胞准备阶段,细胞所有的基因存储寄存器均参与移位.在自修复过程的基因更新阶段,非控制细胞内空闲基因后的寄存器不参与移位,需要寄存器控制电路根据细胞状态及前方寄存器移位状态产生相应的寄存器移位控制信号.

对于细胞中的第  $i$  个基因存储寄存器,其移位使能信号  $R_iSE$  由阵列状态信号  $RunState$ 、细胞控制角色信号  $CE$  和第  $i-1$  个寄存器移位状态  $R_{i-1}SE$  及其基因标识  $r_{i-1}o(0)$  确定,其确定规则如表 2 所示.

当  $i=1$  时,即对于细胞内的第一个基因存储寄存器,其移位使能信号由  $RunState$ 、 $CE$  和存储单元移位使能信号  $C_iSE$  确定.

表 2 寄存器控制电路功能表

输入信号		输出信号		
RunState	CE	$R_{i-1}SE$	$r_{i-1}o(0)$	$R_iSE$
0	x	x	x	1
1	1	x	x	1
1	0	0	x	0
1	0	1	1	1
1	0	1	0	0

则基因寄存器移位控制电路功能为

$$R_iSE = \begin{cases} \overline{RunState} + CE + C_iSE, & i = 1 \\ \overline{RunState} + CE + R_{i-1}SE * r_{i-1}o(0), & i > 1 \end{cases} \quad (5)$$

## 4 分析对比

结合所提基因存储在自修复过程中的特点,对其可靠性、硬件消耗及在大规模通用自修复芯片中的应用进行了分析.分析过程中设阵列规模为  $M \times N$ ,目标电路规模为  $m \times n$ ,基因宽度为  $w$ .

### 4.1 可靠性分析

根据自修复过程的基因更新特征,设每个细胞可靠性符合指数分布,即  $r(t) = e^{-\lambda t}$ ,自修复过程中采用列移除自修复,则系统可靠性函数<sup>[11]</sup>为

$$\begin{aligned} R_{rr}(t) &= R_r(t) * (1 - \sum_{g=k}^n P(t)) \\ &= \sum_{i=n}^N C_N^i R_r(t)^i (1 - R_r(t))^{N-i} \\ &\quad * [1 - \sum_{g=k}^n C_{n-k+1}^{g-k+1} R_r(t)^{n-g} (1 - R_r(t))^g] \end{aligned} \quad (6)$$

其中  $R_r(t) = r(t)^m = e^{-m\lambda t}$  为每一行的可靠度.

系统的失效前平均时间(Mean Time to Failure, MTTF)  $MTTF_{rr}$  为

$$\begin{aligned} MTTF_{rr} &= \int_0^{\infty} R_{rr}(t) dt \\ &= \int_0^{\infty} \sum_{i=n}^N C_N^i R_r(t)^i (1 - R_r(t))^{N-i} \\ &\quad * [1 - \sum_{g=k}^n C_{n-k+1}^{g-k+1} R_r(t)^{n-g} (1 - R_r(t))^g] dt \end{aligned} \quad (7)$$

对于规模为  $20 * 20$  的目标电路,当其在  $20 * 30$ 、 $20 * 40$ 、 $20 * 50$ 、 $20 * 60$ 、 $20 * 70$ 、 $20 * 80$  等不同规模的阵列上实现,且基因备份数目在  $[2, 20]$  变化时,系统 MTTF 的变化如图 5 所示.

由图 5 可以看出,系统的 MTTF 与阵列规模和基因备份数目相关.当阵列规模越大时,系统中冗余行/列数目越大,其 MTTF 越大,这一特性与已有基因存储是一致的.

由于具有基因更新过程,当基因备份数目较小时,系统 MTTF 随基因备份数目的增加而快速升高,当基因备份数目稍大时,系统 MTTF 趋于稳定.

已有的循环备份基因存储和原核基因存储也存在基因更新过程,但其基因备份数目为 2,由图 5 可知,其系统 MTTF 较低.而本文基因存储可以任意设置基因备份数目,通过将基因备份数目设置为 5~10,可以在较小的基因备份数目下取得较大的系统可靠性,使得本文基因存储避免了循环备份基因存储和原核基因存储中系统可靠性低的问题.

### 4.2 硬件消耗分析

本文基因存储的硬件消耗主要包括两部分:基因

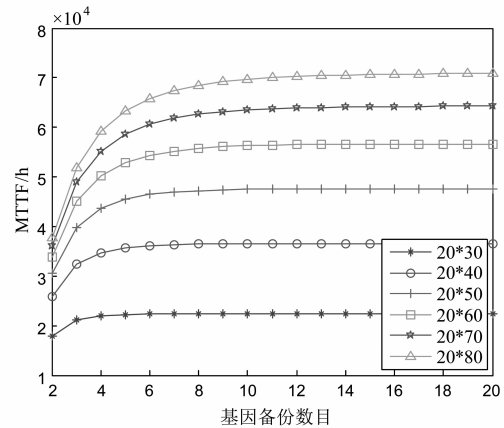


图5 不同阵列、基因备份数目下MTTF

存储和控制电路,首先对基因存储和控制电路在 Xilinx ISE 12.2 中进行了实现,通过综合分析了其硬件消耗,其次根据硬件消耗模型,分析了基因存储的硬件消耗 MOS 管数目.

#### 4.2.1 控制电路硬件消耗分析

为了研究控制电路硬件消耗,在 Xilinx ISE 12.2 中对控制电路和基因存储进行了实现,并通过 XST 进行了电路综合,统计综合后使用 Slice、FF 及 LUT 数目.

通过 3.2 节可知,控制电路只与基因备份数目  $k$  相关,而与基因宽度  $w$  无关;而基因存储与基因备份数目和基因宽度均相关.因此,在不同的基因备份数目、基因宽度下进行了基因存储的实现.设基因备份数目分别为 4、8、16、32、64 和 128,基因宽度分别为 32、64、128,基因存储和控制电路的硬件消耗如表 3 所示.

由表 3 所示不同基因备份数目和基因宽度下存储单元和控制单元硬件消耗可以看出,存储单元的硬件消耗随着基因备份数目和基因宽度的增加而增加,这是由于随着基因备份数目、基因宽度的增加,存储单元中需要的寄存器数目相应增加造成的,因此硬件消耗中 FF 数目随着基因备份数目、基因宽度增加明显,而存储单元中逻辑电路消耗较少,且随基因宽度变化不大,如表 3 中 4-LUT 数目;控制电路的硬件消耗随着基因备份数目的增加而增加,FF 消耗数目较少,而逻辑电路所消耗 4-LUT 数目随着基因备份数目的增加而显著增加,这是由于控制电路中计算式(3)与基因备份数目相关造成的,当基因备份数目较大时,式(3)的计算需要较大的逻辑电路完成.

当基因备份数目较小时,控制电路硬件消耗与基因存储的硬件消耗比较小,如  $k \leq 16$  时,消耗比低于 6.6%,当基因备份数目较大时,控制电路硬件消耗增加,当  $k = 64, w = 32$  时,消耗比达到 16.3%,其余均在 15% 以下.

表 3 基因存储与控制电路的硬件消耗

k	硬件类型	基因存储			控制电路	消耗比
		w = 32	w = 64	w = 128		
4	Slice	64	128	256	2	≤3.1%
	FF	128	256	512	4	
	4-LUT	5	5	5	3	
8	Slice	128	256	512	3	≤2.3%
	FF	256	512	1024	6	
	4-LUT	2	3	14	6	
16	Slice	256	512	1024	17	≤6.6%
	FF	512	1024	2048	6	
	4-LUT	33	33	33	33	
32	Slice	512	1024	2048	49	≤9.6%
	FF	1024	2048	4096	10	
	4-LUT	60	60	60	98	
64	Slice	1024	2048	4096	167	≤16.3%
	FF	2048	4096	8192	10	
	4-LUT	135	136	136	288	
128	Slice	2048	4096	8192	286	≤14.0%
	FF	4096	8192	16384	10	
	4-LUT	274	136	274	481	

通过选择较小的基因备份数目,可以在保持系统可靠性变化不大的情况下降低基因存储的硬件消耗,同时降低控制电路的硬件消耗.当基因备份数目小于16时,控制电路的硬件消耗可以忽略不计.

4.2.2 基因存储消耗分析

本文存储结构基于寄存器实现,使用 MOS 管数目计算基因存储的硬件消耗,其计算模型<sup>[11]</sup>为

$$H = M * N * k * w * 24 \quad (8)$$

对于 5.1 节的目标电路,若阵列基因宽度为 57<sup>[10]</sup>,则硬件消耗如图 6 所示.

由图 6 可以看出,消耗 MOS 管数目随着基因备份数目的增加而线性上升,且阵列规模越大,其上升速度越快.

结合图 5 可以看出,通过选择较小的基因备份数目,可以在保持较大的系统 MTTF 的情况下降低系统的硬件消耗.

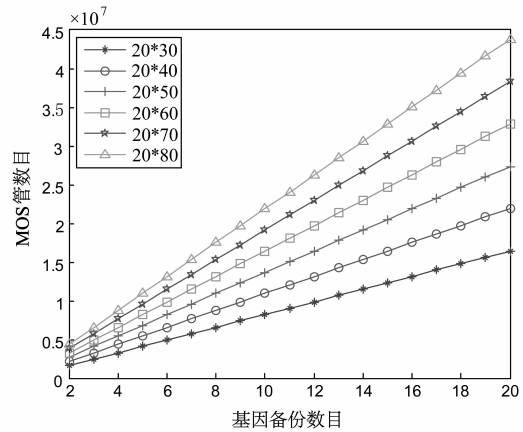


图 6 不同阵列、基因备份数目下硬件消耗

4.3 大规模自修复芯片中的应用分析

仿生电子细胞一个重要研究目的是研制具有自修复能力的通用仿生芯片,其上运行的目标电路规模不确定,有  $1 \leq m \leq M, 1 \leq n \leq N$ . 为了支持不确定规模的目标电路运行,细胞内所需基因存储规模有:

全存储:  $\max(m * n)$ ;

行列存储:  $\max(n)$ ;

部分基因存储:  $\max((M - m + 1) * (N - n + 1))$ ;

循环变形存储:  $\max(n)$ ;

则不同存储用于设计通用自修复芯片时,细胞内及阵列内所需基因存储规模如表 4 所示.

由表 4 可以看出,全存储、行/列存储、部分基因存储和循环变形存储细胞内存储规模与阵列规模相关,芯片中所需基因存储规模与阵列规模呈非线性增加关系;循环备份存、原核存储及本文存储中细胞内存储规模为常数,与阵列规模无关,芯片中所需基因存储规模与阵列规模成线性增长关系.

当芯片规模从  $10 * 10$  到  $100 * 100$  变化,分别采用全存储、行/列存储、部分基因存储、循环变形存储、循环备份存储、原核细胞存储及本文存储时,且采用本文存储时,基因备份数目  $k$  取 10,芯片中存储规模如图 7 所示.

由图 7 可以看出,当芯片行/列数目从 10 增加到 100,即芯片中细胞数目增加 100 倍时,全存储、部分基因存储的规模从  $10^4$  增加到  $10^8$ ,增加了  $10^4$  倍;行/列存储、循环变形存储从  $10^3$  增加到  $10^6$ ,增加了  $10^3$  倍;循环

表 4 芯片中存储规模

基因存储	全存储	行列存储	部分基因存储	循环变形存储	循环备份存储	原核存储	本文存储
细胞内存储规模	$M * N$	$N$	$M * N$	$N$	2	2	$k$
阵列内存储规模	$M * N * (M * N)$	$M * N * N$	$M * N * (M * N)$	$M * N * N$	$M * N * 2$	$M * N * 2$	$M * N * k$

变形存储、原核存储从  $2 * 10^2$  增加到  $2 * 10^4$ , 增加了 100 倍; 本文存储在基因备份数目  $k = 10$  时, 基因存储规模由  $10^3$  增加到  $10^5$ , 增加了 100 倍. 当芯片行/列数目为 100 时, 全存储、部分基因存储, 行/列存储、循环变形存储, 本文存储 ( $k = 10$ ), 循环备份存储、原核存储所需存储规模比例为  $10^8 : 10^6 : 10^5 : 2 * 10^4$ , 即  $10^4 : 10^2 : 10 : 2 = 10^4 : 10^2 : k : 2$ .

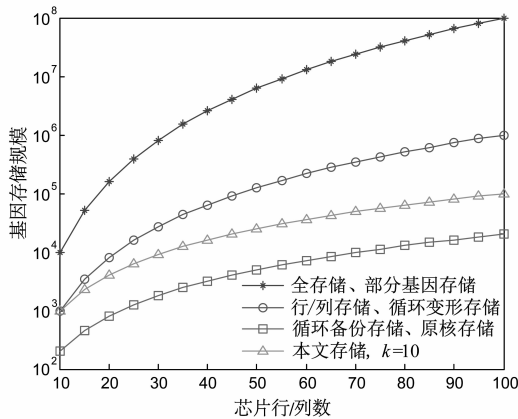


图7 不同芯片规模下所需存储规模

### 5 实验验证

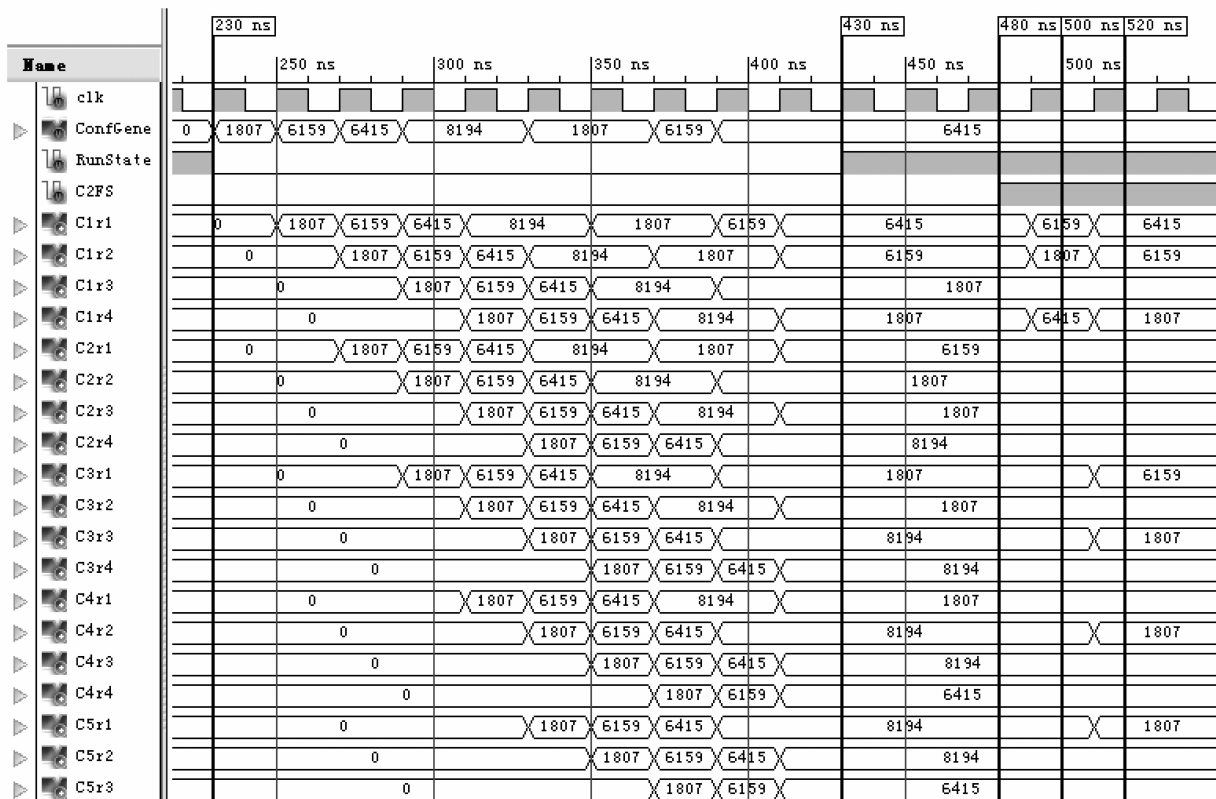
采用本文基因存储结构在 Xilinx ISE12.2 环境中

对文献[4]中的半加器进行实现, 阵列规模为 6 行 5 列, 并利用 ISE 自带的仿真软件 ISim 对功能分化及自修复过程进行仿真.

实验中以第一列细胞为观察对象, 记为 C1、C2、C3、C4、C5、C6, 其表达基因分别为 3207、3079、903、903、4097、4097, 其中 4097 为空闲基因, C5、C6 为空闲细胞. 为了对功能基因和空闲基因进行识别, 需要增加基因标志位, 实际存储基因为 6415、6159、1807、1807、8194、8194. 基因备份数目  $k$  设置为 4. 实验中监测细胞基因存储器内容及关键控制信号,  $Cirj$  为第  $i$  个细胞的第  $j$  个寄存器存储内容;  $CirjCE$  为第  $i$  个细胞内第  $j$  个寄存器的移位控制信号.

#### 5.1 阵列的功能分化及单细胞故障自修复

按照 3.1.1 节基因配置串规则, 分化过程所需输入配置 ConfGene 为 1807、6159、6415、8194、8194、1807、1807、6159、6415. 230ns 阵列的状态控制信号 RunState 置为 0, 开始功能分化. 在时钟 clk 的驱动下, 配置基因依次送入母细胞, 并由母细胞的  $r1$  寄存器开始在阵列上按照 3.1 节规则扩散. 430ns, 细胞表达基因均存入其第一个基因寄存器中, 且细胞内另外三个寄存器存储后续三个细胞的表达基因, 阵列分化完成, 耗时 10 个时钟周期. 具体过程如图 8 所示.



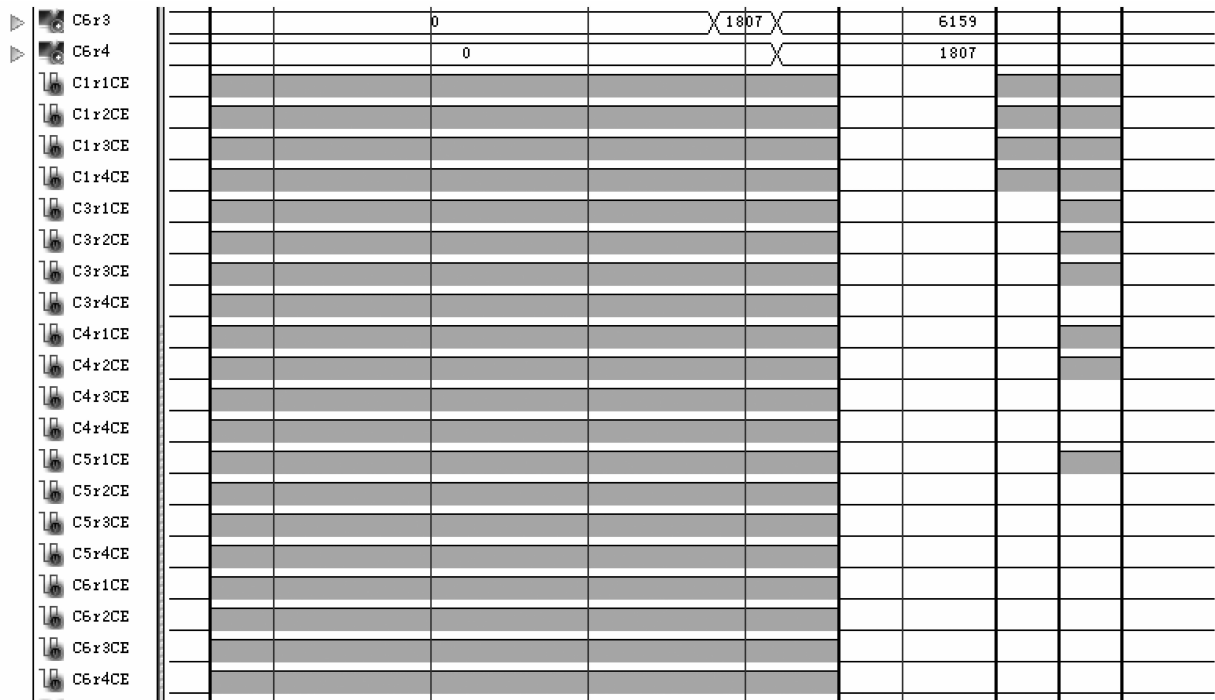


图8 阵列功能分化及C2细胞故障自修复

在阵列运行过程中,480ns C2 细胞故障,自检单元发出细胞故障信号,C2FS 置为高,开始自修复过程:

(1) C1 为修复控制细胞,进行一个周期的逆时针循环移位,如 500ns 所示;

(2) C1 进行一次顺时针循环移位,同时阵列中的 C3、C4、C5 进行一次顺时针非循环移位,如 520ns 所示。

至 520ns,故障细胞 C2 的表达基因 6159 移至 C3 的 r1 寄存器,由 C3 细胞表达,C3、C4 的功能分布由 C4、C5 执行,完成故障的自修复,耗时 2 个时钟周期。

## 5.2 相邻多细胞故障的自修复

480ns 相邻的 C2、C3 细胞故障,细胞故障信号 C2FS、C3FS 置为高. C1 转变为控制细胞,控制完成整个自修复过程,如图 9 所示。

首先,C1 内基因信息逆时针循环移位两次,为基因更新做准备,如 480ns ~ 520ns 所示;然后 C1 和更新范围内细胞同时进行两次顺时针循环移位和顺时针非循环移位,完成基因的更新,如 520ns ~ 560ns 所示。

至 560ns,故障细胞 C2、C3 功能由其后的 C4、C5 执行,C4 的表达基因 1807 移至 C6 细胞的 r1 寄存器,由 C6 表达,阵列完成自修复,耗时 4 个时钟周期。

通过实验可以看出,所设计的部分基因循环存储通过细胞内和细胞间基因移位,能够完成阵列的分化和自修复,且分化、自修复时间较短。

## 6 总结

设计了一种新型的基因存储方式—部分基因循环

存储,采用了相关冗余进行基因存储,细胞只存储系统的部分基因,自修复过程中通过相邻细胞的基因存储进行基因更新,更新过程中无需故障细胞参与,可适用于细胞的多种故障的自修复。

模仿生物的 HTG 和 IGT 现象,所提出的基因存储通过细胞间、细胞内的基因移位进行阵列的功能分化和自修复,分化耗时(细胞个数 + 基因备份数目)个时钟周期,自修复耗时(故障细胞数目 \* 2)个时钟周期,分化过程迅速,而自修复时耗与故障细胞数目相关,耗时极短。

该基因存储中基因备份数目与目标电路和阵列的规模无关,通过选择基因备份数目,可在保持系统可靠性的基础上降低硬件消耗. 且系统中基因存储规模与阵列规模成线性关系增加,可用于大规模通用芯片的设计。

## 参考文献

- [1] 郝国峰,王友仁,张岩,等. 可重构硬件芯片级故障定位与自主修复方法[J]. 电子学报,2012,40(2):384-388. HAO Guo-feng, WANG You-ren, ZHANG Zhai, et al. In-chip fault localization and self-repairing method for reconfigurable hardware [J]. Acta Electronica Sinica, 2012, 40 (2):384-388. (in Chinese)
- [2] A H Jackson, R Canham, A M Tyrrell. Robot fault-tolerance using an embryonic array [A]. Proceedings of the 2003 NASA/Dod Conference on Evolvable Hardware[C].

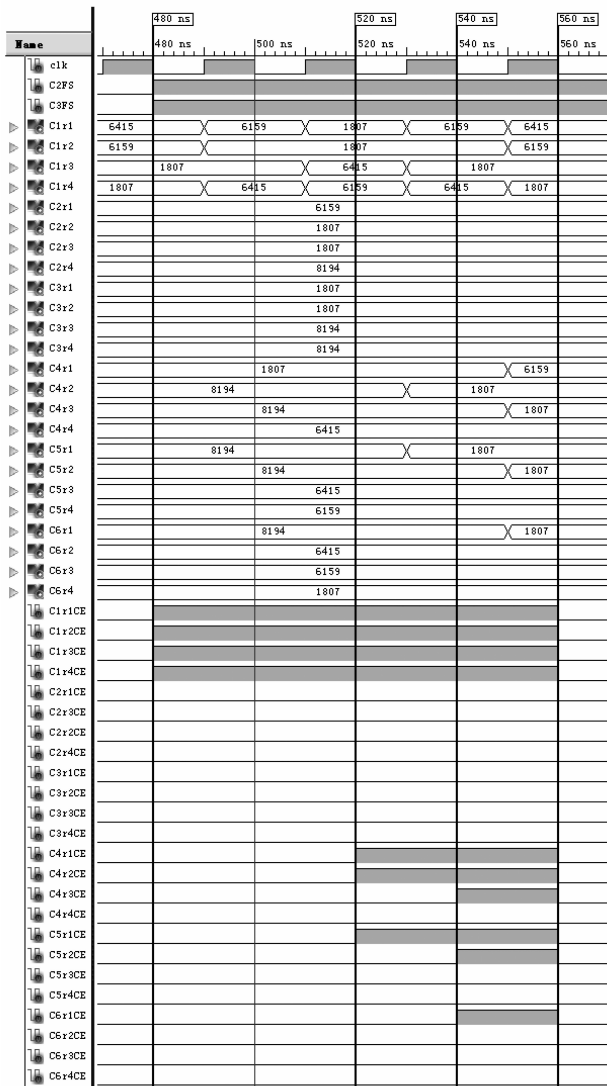


图9 C2、C3故障修复过程

Los Alamitos:IEEE Comput Soc,2003,91 – 100.

[3] M Samie,E Farjah,G Dragffy. Cyclic metamorphic memory for cellular bio-inspired electronic systems [ J ]. Genet Program Evol M,2008,9(3) :183 – 201.

[4] X Zhang,G Dragffy,A G Pipe, et al. Partial-DNA supported artificial-life in an embryonic array[ A ]. Proceedings of the International Conference on VLSI [ C ]. Avon: Csrea Pressathens,2004,203 – 208.

[5] Xu Guili,Xia Zhenghao,Wang Haibin, et al. Design of embryo-electronic systems capable of self-diagnosing and self-healing and configuration control [ J ]. Chinese Journal of Aeronautics,2009,22(6) :637 – 643.

[6] 李岳,王南天,钱彦岭. 原核细胞仿生自修复电路设计 [ J ]. 国防科技大学学报,2012,34(3) :154 – 157.  
LI Y,WANG N T,QIAN Y L. Self-healing circuit design inspired by prokaryotic cell[ J ]. Journal of National Univer-

sity of Defense Technology,2012,34(3) :154 – 157 ( in Chinese ).

[7] M Samie,G Dragffy,A M Tyrrell, et al. Novel bio-inspired approach for fault-tolerant VLSI systems[ J ]. IEEE T VLSI Syst,2012,21(10) :1878 – 1891.

[8] 王洽,乐霖培,张体操,等. 水平基因转移在生物进化中的作用. 科学通报,2014,59:2055 – 2064.  
WANG Q,YUE J P,ZHANG T C, et al. Horizontal gene transfer provides new insights into biological evolution[ J ]. Chin Sci Bull ( Chin Ver ),2014,59:2055 – 2064. ( in Chinese )

[9] 张瑞福,蒋建东,代先祝,等. 环境中污染物降解基因的水平转移(HGT)及其在生物修复中的作用[ J ]. 遗传,2005,27(5) :845 – 851.  
ZHANG Rui-fu,JIANG Jian-dong,DAI Xian-zhu, et al. Horizontal transfer of environmental pollutant-degrading gene and application in bioremediation [ J ]. Hereditas,2005,27(5) :845 – 851. ( in Chinese )

[10] S Zhu,J Y Cai,Y F Meng, et al,A novel structure of embryonics electronic cell array[ J ]. WSEAS Trans Cir Syst,2014,13:224 – 232.

[11] 朱赛,蔡金燕,孟亚峰,等. 胚胎电子细胞中基因备份数目优选方法[ J/OL ]. 北京航空航天大学学报,http://www.cnki.net/kcms/detail/11.2625.V.20150617.1132.003.html,2015-06-20.  
Zhu S,Cai J Y,Meng Y F, et al. Gene backup number selection method for embryonics cell[ J/OL ]. Journal of Beijing University of Aeronautics and Astronautics, http://www.cnki.net/kcms/detail/11.2625.V.20150617.1132.003.html,2015-06-20. ( in Chinese )

作者简介



蔡金燕 女,1961 年生于天津.教授,博士生导师.研究方向为电子系统性能检测与故障诊断、电子系统可靠性分析与设计及电子系统仿生自修复设计.  
E-mail:radarc@126.com



朱赛 男,1987 年生于河南郸城.博士研究生,研究方向为电子系统仿生自修复设计.