



(12) 发明专利

(10) 授权公告号 CN 1967719 B

(45) 授权公告日 2011.08.17

(21) 申请号 200610148491.X

(56) 对比文件

(22) 申请日 2006.11.17

US 5920201 A, 1999.07.06, 全文.

US 6114843 A, 2000.09.05, 全文.

(30) 优先权数据

US 6232893 B1, 2001.05.15, 全文.

60/737, 868 2005.11.17 US

11/335, 437 2006.01.18 US

审查员 杭雪蒙

(73) 专利权人 奥特拉有限公司

地址 美国加利福尼亚

(72) 发明人 刘令时 陈天沐

(74) 专利代理机构 北京三友知识产权代理有限公司 11127

代理人 李辉

(51) Int. Cl.

G11C 11/412(2006.01)

G11C 11/413(2006.01)

G11C 8/16(2006.01)

G06F 13/16(2006.01)

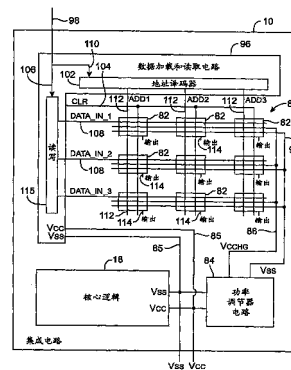
权利要求书 1 页 说明书 15 页 附图 13 页

(54) 发明名称

电源电平升高的可编程逻辑器件存储器单元

(57) 摘要

本发明提供了电源电平升高的可编程逻辑器件存储器单元。可编程逻辑器件集成电路包含被按可编程核心逻辑电源电压供电的可编程核心逻辑。向存储器单元中加载可编程逻辑器件配置数据,来对可编程核心逻辑进行配置以执行定制逻辑功能。在正常操作过程中,可以按比可编程核心逻辑电源电压高的电源电压对存储器单元供电。在数据加载操作过程中,可以按与可编程核心逻辑电源电压相等的电源电压对存储器单元供电。数据加载和读取电路向存储器单元中加载数据并从存储器单元读取数据。该数据加载和读取电路生成地址信号。地址信号在数据写入操作过程中可以具有比在读取操作过程中高的电压电平。



1. 一种集成电路,该集成电路包括:

可编程核心逻辑,其被按可编程核心逻辑电源电压供电;

多个存储器单元,所述存储器单元具有对所述可编程核心逻辑进行配置的输出;

多条地址线,其用于对所述存储器单元进行寻址;以及

数据加载和读取电路,其向所述地址线施加具有比所述可编程核心逻辑电源电压高的电压电平的地址信号,以把配置数据加载到所述存储器单元中。

2. 根据权利要求1所述的集成电路,该集成电路还包括:

功率调节器电路,其提供对所述存储器单元供电的时变存储器单元电源电压,其中,当所述数据加载和读取电路向所述地址线施加所述地址信号以加载所述配置数据时,所述时变存储器单元电源电压等于所述可编程核心逻辑电源电压,并且其中,在正常操作过程中,所述时变存储器单元电源电压高于所述可编程核心逻辑电源电压。

3. 根据权利要求1所述的集成电路,该集成电路还包括功率调节器电路,该功率调节器电路在所述存储器单元的正常操作过程中按高于所述可编程核心逻辑正电源电压的存储器单元电源电压对所述存储器单元供电。

4. 根据权利要求1所述的集成电路,其中,每个所述存储器单元都包括多个交叉耦合反相器,并且其中,所述数据加载和读取电路向所述地址线施加地址信号,以从所述存储器单元读取配置数据,其中,用来读取所述配置数据的地址信号所具有的电压电平,低于在将所述配置数据加载到所述存储器单元中时使用的地址信号的电压电平。

5. 根据权利要求1所述的集成电路,该集成电路还包括多个地址晶体管,其中,每个所述地址晶体管都与所述多个存储器单元中的相应存储器单元相关联并且具有栅极,其中,所述栅极由所述地址线来控制,其中,在将所述配置数据加载到所述存储器单元中的数据写入操作过程中,所述栅极接收第一电压电平的地址信号,而在由所述数据加载和读取电路从所述存储器单元读取所述配置数据的数据读取操作过程中,所述栅极接收第二电压电平的地址信号,其中,所述第一电压电平高于所述可编程核心逻辑电源电压,并且其中,所述第二电压电平等于所述可编程核心逻辑电源电压。

6. 根据权利要求1所述的集成电路,该集成电路还包括:

多个地址晶体管,其中,每个所述地址晶体管都与所述多个存储器单元中的相应存储器单元相关联并且具有栅极,其中,所述栅极由所述地址线来控制,其中,在将所述配置数据加载到所述存储器单元中的数据写入操作过程中,所述栅极接收第一电压电平的地址信号,而在由所述数据加载和读取电路从所述存储器单元读取所述配置数据的数据读取操作过程中,所述栅极接收第二电压电平的地址信号,其中,所述第一电压电平高于所述可编程核心逻辑电源电压,其中,所述第二电压电平等于所述可编程核心逻辑电源电压;和

功率调节器电路,其在所述存储器单元的正常操作过程中,按等于所述第一电压电平的电源电压对所述存储器单元供电。

## 电源电平升高的可编程逻辑器件存储器单元

### 技术领域

[0001] 本发明涉及存储器单元,更具体来说,涉及具有针对诸如可编程逻辑器件的集成电路的升高输出电压的易失性存储器单元。

### 背景技术

[0002] 集成电路往往包含多个易失性存储器单元。典型的易失性存储器单元基于交叉耦合反相器(锁存器)并用于存储数据。每个存储器单元都可以存储一比特数据。

[0003] 通常使用易失性存储器单元来存储可编程逻辑器件中的配置数据。可编程逻辑器件是如下类型的集成电路:其可以按相对小的批量来定制以实现希望的逻辑设计。在典型的场合下,可编程逻辑器件制造商预先设计并制造非定制的可编程逻辑器件集成电路。随后,逻辑设计者使用逻辑设计系统来设计定制逻辑电路。该逻辑设计系统利用与制造商的可编程逻辑器件的硬件性能有关的信息,来帮助设计者利用给定可编程逻辑器件上可用的资源实现逻辑电路。

[0004] 逻辑设计系统基于逻辑设计者的定制设计来创建配置数据。当将该配置数据加载到一个可编程逻辑器件的存储器单元中时,该配置数据对该可编程逻辑器件的逻辑进行编程,使得该可编程逻辑器件实现设计者的逻辑电路。使用可编程逻辑器件可以显著减小实现希望的集成电路设计所需的工作量。

[0005] 按正电源电压向常规可编程逻辑器件存储器单元供电。用来向常规可编程逻辑器件存储器单元供电的该正电源电压,通常被称为  $V_{cc}$  或  $V_{cc-core}$ ,并且同样是用来向可编程逻辑器件中的核心逻辑供电的电源电压。

[0006] 按低  $V_{cc}$  值操作的诸如可编程逻辑器件集成电路的集成电路,提供优于按较高  $V_{cc}$  值操作的集成电路的有益效果。例如, $V_{cc}$  的降低通常导致功耗的降低。由于这些有益效果,半导体产业不断地努力提出支持降低  $V_{cc}$  的工艺和电路设计。前几代可编程逻辑器件按 2.0 伏特、1.8 伏特以及 1.5 伏特的  $V_{cc}$  电平操作。较近以来,已在可编程逻辑器件中使用了 1.2 伏特的  $V_{cc}$  电平。预期将来的可编程逻辑器件将支持低于 1.2 伏特(例如,1.1 伏特或 1.0 伏特)的  $V_{cc}$  电平。

[0007] 可编程逻辑器件中的存储器单元产生反映已加载到这些存储器单元中的配置数据的静态输出信号。这些静态输出信号对 n 沟道和 p 沟道金属氧化物半导体(MOS)晶体管的栅极进行驱动。这些晶体管中的某些晶体管(如 n 沟道晶体管)用作传输晶体管,并被安装到复用器和其他逻辑组件中。有时 p 沟道晶体管用作防止向集成电路的未用部分施加电力的断电晶体管。当按不足的电压驱动 n 沟道晶体管和 p 沟道晶体管时,它们都操作得很差。例如,如果 n 沟道传输晶体管的栅极接收到太低的电压,则该晶体管不会正确地导通,而会使经过该晶体管的逻辑信号劣化。如果 p 沟道断电晶体管的栅极电压太低,则该晶体管不会正确地截止,而会出现不期望的大泄漏电流。

[0008] 因此,期望能够提供在低核心逻辑电源电平下操作良好的可编程逻辑器件集成电路。

## 发明内容

[0009] 根据本发明,提供了包含可编程核心逻辑的可编程逻辑器件集成电路。将配置数据加载到存储器单元阵列中。被加载的存储器单元产生对核心逻辑进行配置以产生定制逻辑功能的控制信号。

[0010] 使用数据加载和读取电路来对存储器单元阵列中的存储器单元进行寻址。为了减小与针对存储器单元的地址晶体管相关联的占用面积 (real estate) 需求并且 / 或者增大存储器单元的写入容限 (write margin), 在数据写入操作过程中使用的地址信号可以具有比可编程核心逻辑电源电压高的电压。在读取操作过程中, 可以按诸如可编程核心逻辑电源电压的较低电压向存储器单元施加地址信号。

[0011] 在正常操作过程中, 可以使用升高的电源电压来向存储器单元供电。在数据加载操作过程中, 可以将针对存储器单元的电源电平降低到可编程核心逻辑电源电压。

[0012] 本发明提供了一种集成电路, 该集成电路包括: 可编程核心逻辑, 其被按可编程核心逻辑电源电压供电; 多个存储器单元, 所述存储器单元具有对所述可编程核心逻辑进行配置的输出; 多条地址线, 其用于对所述存储器单元进行寻址; 以及数据加载和读取电路, 其向所述地址线施加具有比所述可编程核心逻辑电源电压高的电压电平的地址信号, 以把配置数据加载到所述存储器单元中。

[0013] 根据附图和以下详细说明, 将更加清楚本发明的其他特征、本发明的特性和各种优点。

## 附图说明

[0014] 图 1 是根据本发明的例示性可编程逻辑器件集成电路的图。

[0015] 图 2 是常规的可编程逻辑器件存储器单元阵列的图。

[0016] 图 3 是常规可编程逻辑器件存储器单元的图。

[0017] 图 4 是示出如何对常规可编程逻辑器件存储器单元加载具有逻辑 1 值的配置数据比特的定时图。

[0018] 图 5 是示出如何对常规可编程逻辑器件存储器单元加载具有逻辑 0 值的配置数据比特的定时图。

[0019] 图 6 是根据本发明的带有存储器单元电路的例示性可编程逻辑器件的图。

[0020] 图 7 是根据本发明的针对集成电路 (如图 6 所示类型的可编程逻辑器件集成电路) 的例示性存储器单元的图。

[0021] 图 8 是根据本发明的例示性读写电路的电路图。

[0022] 图 9 是示出根据本发明如何对图 7 所示类型的可编程逻辑器件存储器单元加载具有逻辑 1 值的配置数据比特的定时图。

[0023] 图 10 是示出根据本发明如何对图 7 所示类型的可编程逻辑器件存储器单元加载具有逻辑 0 值的配置数据比特的定时图。

[0024] 图 11 是示出根据本发明如何读取图 7 所示类型的可编程逻辑器件存储器单元以对正确加载了具有逻辑 1 值的配置数据比特进行确认的定时图。

[0025] 图 12 是示出根据本发明如何读取图 7 所示类型的可编程逻辑器件存储器单元以

对正确加载了具有逻辑 0 值的配置数据比特进行确认的定时图。

[0026] 图 13 是示出根据本发明对图 7 所示类型的可编程逻辑器件存储器单元的阵列进行使用时涉及的例示性步骤的流程图。

### 具体实施方式

[0027] 本发明涉及存储器单元和包含存储器单元的集成电路。本发明还涉及用于将数据加载到存储器单元中和用于确认正确地加载了数据的电路。包含存储器单元的集成电路可以是存储器芯片、带有存储器阵列的数字信号处理电路、微处理器、带有存储器阵列的专用集成电路、其中针对配置存储器使用了存储器单元的可编程逻辑器件集成电路、或者任何其他合适的集成电路。为清楚起见，总体上来说，将在其中使用易失性存储器单元来存储配置数据的可编程逻辑器件集成电路的背景下，对本发明进行描述。

[0028] 在对可编程逻辑器件进行编程的过程中，将配置数据加载到存储器单元中。在可编程逻辑器件的操作过程中，每个存储器单元都提供静态输出信号。来自存储器单元的输出信号被施加给可编程逻辑并对可编程逻辑进行定制以执行期望的逻辑功能。在典型布置中，每个静态输出信号都充当施加给 n 沟道或 p 沟道金属氧化物半导体晶体管的栅极的控制信号。

[0029] 包含有 n 沟道和 p 沟道晶体管的可编程逻辑构成了可编程逻辑器件的逻辑核心的一部分，因此有时被称为可编程核心逻辑。使用通常称为  $V_{cc}$  的正电源电压和通常称为  $V_{ss}$  的接地电压来向核心逻辑供电。

[0030] 为了降低功耗，在半导体产业中存在降低  $V_{cc}$  的大小的发展趋势。降低核心逻辑电源电压会降低功耗，但是会带来设计挑战。例如，可能难以使可编程核心逻辑中的 n 沟道晶体管完全导通，并且可能难以使核心逻辑中的 p 沟道晶体管完全截止。

[0031] 根据本发明，可以使用升高的电源电平来向可编程逻辑器件集成电路上的可编程存储器单元供电。该升高的电源电平使用正电源电压  $V_{cchg}$  和接地电压。对于一个合适的布置，接地电源电压是  $V_{ss}$ ，这同样是在核心逻辑中使用的值。

[0032] 在器件正常操作过程中，正电源电压  $V_{cchg}$  保持为值  $V_{cchg-high}$ 。结果，包含逻辑 1 的存储器单元按  $V_{cchg-high}$  产生输出信号。值  $V_{cchg-high}$  大于  $V_{cc}$ ，因此与按  $V_{cc}$  向存储器单元供电时可能会出现的情况相比，更充分地使可编程核心逻辑中的 n 沟道晶体管导通，并且更充分地使可编程核心逻辑中的 p 沟道晶体管截止。

[0033] 图 1 示出了根据本发明的例示性可编程逻辑器件 10。

[0034] 可编程逻辑器件 10 可以具有用于通过输入 / 输出插脚 14 将信号从器件 10 驱动出去并从其他器件接收信号的输入 / 输出电路 12。可以使用互连资源 16，如全局和局部的垂直和水平导线和总线，来在器件 10 上对信号进行择路。互连资源 16 包括固定互连（导线）和可编程互连（即，相应固定互连之间的可编程连接）。可编程逻辑 18 可以包括组合和顺序逻辑电路。可以将可编程逻辑 18 构造成执行定制逻辑功能。可以将与互连资源相关联的可编程互连视为可编程逻辑 18 的一部分。

[0035] 可编程逻辑器件 10 包含易失性存储器单元 20，可以使用插脚 14 和输入 / 输出电路 12 向所述易失性存储器单元 20 加载配置数据（也称为编程数据）。一旦加载了配置数据，这些存储器单元中的每一个就都提供对可编程逻辑 18 中的关联逻辑组件的状态进行

控制的对应静态控制输出信号。典型的是,使用这些存储器单元输出信号来控制金属氧化物半导体 (MOS) 晶体管的栅极。这些晶体管中的大部分通常是诸如复用器的可编程组件中的 n 沟道金属氧化物半导体 (NMOS) 传输晶体管。当存储器单元输出为高电平时,由该存储器单元控制的传输晶体管导通,并将逻辑信号从它的输入传输到它的输出。当存储器单元输出为低电平时,该传输晶体管截止,因而不传输逻辑信号。

[0036] 典型的存储器单元 20 由被设置成形成交叉耦合反相器的多个晶体管构成。根据一个合适的方法,使用互补金属氧化物半导体 (CMOS) 集成电路技术来形成存储器单元 20,因此这里作为示例对 CMOS 型存储器单元实现进行描述。在可编程逻辑器件集成电路的背景下,存储器单元存储配置数据,因此有时被称为配置随机存取存储器 (CRAM) 单元。

[0037] 可以从任何合适的源对存储器单元进行加载。对于一个例示性布置,通过插脚 14 和输入 / 输出电路 12,从称为配置器件的外部可擦除可编程只读存储器和控制芯片对存储器单元进行加载。将来自被加载的存储器单元 20 的输出信号施加给可编程逻辑 18 中的电路部件 (例如,金属氧化物半导体晶体管) 的端子 (例如,栅极),以对这些部件进行控制 (例如,使某些晶体管导通或截止),从而对可编程逻辑 18 中的逻辑进行配置。这些电路部件可以是诸如传输晶体管的晶体管、复用器的部分、查找表、逻辑阵列、“与”逻辑门、“或”逻辑门、“与非”逻辑门以及“或非”逻辑门等。

[0038] 通常将存储器单元 20 布置成阵列图案。在典型的现代可编程逻辑器件中,每个芯片上可能有几百万个存储器单元 20。在编程操作过程中,向存储器单元阵列提供配置数据。一旦加载了配置数据,存储器单元 20 就对可编程逻辑 18 中的电路的各个部分进行选择控制 (例如,导通和截止),从而对它的功能进行定制,使得它按照需要进行操作。

[0039] 可以利用任何合适的架构对器件 10 的电路进行组织。作为示例,可以将可编程逻辑器件 10 的逻辑组织成各包含多个较小逻辑区的多个较大可编程逻辑区的一系列行和列。可以通过诸如关联的垂直和水平导体的互连资源 16 来将器件 10 的逻辑资源互连。这些导体可以包括基本上跨布整个器件 10 的全局导线、诸如跨布器件 10 的一部分的半线 (half-line) 或四分之一线 (quarter line) 的部分线 (fractional line)、具有特定长度 (例如,足以互连几个逻辑区) 的交错线、较短的本地线、或任何其他合适的互连资源布置。如果需要,可以将器件 10 的逻辑布置成更多级或层,在这些级或层中,将多个大区互连以形成还要大的逻辑部分。还有其他器件布置可以使用不按行和列布置的逻辑。

[0040] 当按阵列布置存储器单元时,可以使用水平和垂直导体以及关联的加载电路来向这些存储器单元加载配置数据。图 2 示出了常规数据加载布置。图 2 的布置具有存储器单元 24 的  $3 \times 3$  阵列 22。(实际的存储器阵列通常具有几百或几千行、列,但是使用  $3 \times 3$  阵列作为示例)。阵列 22 通过正电源线 40 和地线 38 来接收电力。线 40 上的电压  $V_{cc}$  通常是 1.2 伏特,地线 38 上的电压  $V_{ss}$  通常是 0 伏特。

[0041] 使用清零线 36 (标记为 CLR) 来对存储器阵列 22 的内容进行清零。在对阵列进行了清零之后,可以加载配置数据。

[0042] 通过输入 32 串行地向寄存器 30 提供配置数据。然后通过 DATA\_IN\_1、DATA\_IN\_2 以及 DATA\_IN\_3 线 26 将配置数据并行地提供给阵列 22。地址译码器 34 通过输入 44 接收寻址信息。作为响应,地址译码器宣称 (assert) 多条地址线 28 中的期望的一条 (即,ADD1、ADD2 或 ADD3)。当宣称了给定列中的地址线时,将数据线 26 上的数据加载到该列中的存储

器单元 24 中。通过系统地对阵列的每一列中的存储器单元进行加载,来填充该阵列。在完全对阵列加载了配置数据之后,各存储器单元 24 的输出 42 产生用于对可编程逻辑器件上的传输晶体管的栅极或其他逻辑组件进行控制的对应静态控制信号。

[0043] 图 3 示出了在图 2 的阵列 22 中所用类型的常规存储器单元 24。如图 3 所示,存储器单元 24 由两个交叉耦合反相器(反相器 46 和反相器 52)形成。反相器 46 具有 p 沟道金属氧化物半导体(PMOS)晶体管 48 和 n 沟道金属氧化物半导体(NMOS)晶体管 50。反相器 52 具有 PMOS 晶体管 54 和 NMOS 晶体管 56。在清零操作过程中,通过激活清零线 36,使 NMOS 晶体管 60 导通。这会将节点 N2 连接到地 38 并将存储器单元 24 清零。通过节点 N2 上的信号来确定存储器单元在线 42(DATA\_OUT)上的输出。

[0044] 当地址线 28 取高电平时,使 NMOS 晶体管 58 导通,并将数据线 26 上的信号驱动到存储器单元 24 中。如果数据线 26 上的信号是高电平,则节点 N1 保持高电平,而存储器单元 24 保持在它的低电平(清零)状态中。输出 DATA\_OUT 是低电平。如果数据线 26 上的信号是低电平,则节点 N1 取低电平,并且由于反相器 46 使低电平 N1 信号反相,因此节点 N2 上的电压取高电平。这使得输出 DATA\_OUT 是高电平。

[0045] 将 DATA\_OUT 信号施加给传输晶体管 64 的栅极 62。当 DATA\_OUT 是低电平时,使传输晶体管 64 截止。当 DATA\_OUT 是高电平时,使传输晶体管 64 导通并允许数据在线 66 与线 68 之间流动。

[0046] 图 4 和 5 示出了示出与对常规存储器单元 24 进行加载相关联的操作的信号定时图。图 4 中的迹线例示了将逻辑“1”加载到清零后的存储器单元 24 中的过程。图 5 中的迹线例示了将逻辑“0”加载到清零后的存储器单元 24 中的过程。

[0047] 图 4 中的第一条迹线示出了施加给 Vcc 线 40 的电压 Vcc 恒定为 1.2 伏特。将该正电源电压施加给反相器 46 和 52。

[0048] 图 4 中的第二条迹线示出了在系统启动过程中执行了清零操作之后,线 36 上的清零信号 CLR 恒定为 0 伏特。

[0049] 如图 4 中的第三条迹线所示,在时间  $t_1$  处宣称地址线 ADD,而在时间  $t_2$  处去宣称(deassert)地址线 ADD。

[0050] 图 4 中的第四条迹线示出了在数据加载操作过程中线 26 上的 DATA\_IN 的值是恒定的低电平信号。

[0051] 图 4 中的第五条迹线中的信号 N1 表示图 3 中的节点 N1 处的电压。

[0052] 图 4 中的第六条迹线中的信号 DATA\_OUT 与节点 N2 处的电压相同,并且表示存储器单元 24 的内容。当该存储器单元正存储逻辑 1 时,节点 N2 是高电平,并且 DATA\_OUT 是高电平。当该存储器单元正存储逻辑 0 时,节点 N2 是低电平,并且 DATA\_OUT 是低电平。

[0053] 如图 4 所示,在  $t_1$  之前的时间  $t$  处,ADD 是低电平,因此晶体管 58 截止。节点 N1 处的电压是高电平,并且节点 N2 处的电压是低电平。输出 DATA\_OUT 是低电平。在此情况下,存储在存储器单元中的数据是逻辑 0,因为该存储器单元处于其清零状态。在时间  $t_1$  处,ADD 变成高电平,这使晶体管 58 导通并将处于 0 伏特的 DATA\_IN 线连接到节点 N1,使 N1 取低电平。由反相器 46 将节点 N1 上的低电平电压反相,使得节点 N2 上的电压变成高电平。图 4 中的第六条迹线示出这使 DATA\_OUT 在  $t_1$  处变成高电平。在此阶段,在存储器单元 24 中存储了逻辑 1。当在时间  $t_2$  处去宣称地址线 ADD 时,使晶体管 58 截止,这将该存储器单

元隔离并防止了进一步的状态变化。如图 4 中的第六条迹线所示, DATA\_OUT 信号在时间  $t_2$  处保持为高电平。

[0054] 图 5 中的迹线例示了将逻辑“0”加载到已被清零的存储器单元 24(图 3)中的过程。在此情况下, 存储器单元包含逻辑 0, 因此该加载过程不会改变它的状态。

[0055] 图 5 中的第一条迹线示出了施加给  $V_{cc}$  线 40(图 3) 的电压  $V_{cc}$  恒定为 1.2 伏特。将该正电源电压  $V_{cc}$  施加给反相器 46 和 52。

[0056] 如图 5 中的第二条迹线所示, 在系统启动过程中执行了清零操作之后, 线 36 上的清零信号 CLR 恒定为 0 伏特。

[0057] 如图 5 中的第三条迹线所示, 在时间  $t_1$  处宣称地址线 ADD, 接着在时间  $t_2$  处去宣称地址线 ADD。

[0058] 图 5 中的第四条迹线示出了在数据加载操作过程中线 26 上的 DATA\_IN 的值是恒定的高电平值。在图 5 的场合下, 正在向存储器单元 24 中加载逻辑零, 因此 DATA\_IN 是高电平。在图 4 的场合下, 正在向存储器单元 24 中加载逻辑 1, 因此 DATA\_IN 是低电平。

[0059] 图 5 中的第五条迹线中的信号 N1 表示图 3 中的节点 N1 处的电压。当向已被清零的存储器单元中加载零时, N1 不会变化。

[0060] 图 5 中的第六条迹线中的信号 DATA\_OUT(其与节点 N2 处的电压相同) 表示存储器单元 24 的内容。由于对存储器单元 24 进行了清零, 所以在时间  $t_1$  之前 DATA\_OUT 是低电平。在时间  $t_1$  之后, 宣称 ADD, 这使晶体管 58 导通并将高电平 DATA\_IN 信号连接到节点 N1。N1 已经是高电平, 因此向节点 N1 施加高电平 DATA\_IN 信号不会导致存储器单元 24 的状态在时间  $t_1$  处变化, 如图 5 中的 DATA\_OUT 迹线所示。在时间  $t_2$  处去宣称 ADD 线之后, DATA\_OUT 信号也保持固定为它的低电平值。

[0061] 在诸如图 3 的常规存储器单元 24 的常规存储器单元中, 地址信号、清零信号以及数据信号从逻辑低电平值 0 伏特变化到逻辑高电平值  $V_{cc}$ 。

[0062] 根据本发明, 可以使用针对这些信号的不同值来改进性能。例如, 在结合图 4 和 5 描述的类型的数据写入操作过程中可以升高地址信号, 而在数据读取操作过程中可以降低地址信号。此外, 可以升高存储器单元电源电平。

[0063] 如图 6 所示, 根据本发明的可编程逻辑器件集成电路 10 包含数据加载和读取电路 96。将多个存储器单元 82 布置成阵列 88。使用地址线 112 和数据线 108 对该阵列进行寻址。在准备进行数据加载操作时, 使用清零线 104 上的清零信号 CLR 来对存储器单元 82 进行清零。每个存储器单元 82 都在对应的数据输出线 114 上产生输出。

[0064] 可编程逻辑器件 10 包含核心逻辑 18。核心逻辑 18 包含被按正电源电压  $V_{cc}$  和接地电压  $V_{ss}$  供电的可编程核心逻辑。根据一个例示性布置, 正电源电压  $V_{cc}$  约为 1.2 伏特, 接地电压  $V_{ss}$  为 0 伏特。可以使用其他布置。例如,  $V_{cc}$  可以高于或低于 1.2 伏特, 而  $V_{ss}$  可以高于或低于 0 伏特。使用 1.2 伏特的  $V_{cc}$  值和 0 伏特的  $V_{ss}$  值仅仅是例示性的。

[0065] 可以使用任何合适的布置向器件 10 上的电路提供诸如  $V_{cc}$  和  $V_{ss}$  的电源电压。例如, 可以从一个或更多个外部电源插脚接收电源电压。可以使用片上电路来生成比外部提供的电源电压高或低的电源电压。例如, 可以使用分压器来降低外部提供的电压, 并且可以使用电荷泵或其他升压电路来生成比外部提供的电源电压高的电源电压。

[0066] 在图 6 的例示性布置中, 向功率调节器电路 84 提供了外部电源电压  $V_{cc}$  和  $V_{ss}$ 。



功率调节器电路 84 利用这些电压来生成针对阵列 88 中的存储器单元 82 的电源电压  $V_{ss}$  和  $V_{cchg}$ 。

[0067] 数据加载和读取电路 96 向阵列 88 提供地址信号、清零信号以及数据信号。根据一个合适的布置,数据加载和读取电路 96 按高于  $V_{cc}$  的电压来提供这些信号中的至少一些信号。可以从任何合适的源(例如,插脚或功率调节器电路 84)接收这些升高的电压。在图 6 的示例中,数据加载和读取电路 96 根据通过线 85 从外部插脚接收到的电源电压  $V_{ss}$  和  $V_{cc}$  在内部生成升高的电压。还将电源电压  $V_{ss}$  和  $V_{cc}$  提供给核心逻辑 18。

[0068] 核心逻辑 18 包含 n 沟道和 p 沟道金属氧化物半导体晶体管和由 n 沟道和 p 沟道金属氧化物半导体晶体管形成的可编程逻辑组件,如反相器、复用器、逻辑门、寄存器逻辑、查找表等。在典型场合下,可编程核心逻辑 18 中的电路或组件具有接收正电源电压  $V_{cc}$  的正电源端子和接收接地电源电压  $V_{ss}$  的接地电源端子。

[0069] 功率调节器电路 84 利用线 90 和 86 向阵列 88 中的存储器单元 82 提供接地电源电压  $V_{ss}$  和正电源电压  $V_{cchg}$ 。通常,可以使用任何合适的电源电压来对存储器单元 82 供电。当升高电源电压时(即,当正电源电压与接地电压之差大于  $V_{cc}$  时),会增强器件 10 的性能。具体来说,与不使用升高的电源电压的情况相比,使用升高的电源电压使得来自存储器单元 82 的输出控制信号可以更充分地使核心逻辑 18 中的 n 沟道晶体管导通,并且更充分地使核心逻辑 18 中的 p 沟道晶体管截止。

[0070] 通常,可以利用升高的正电源电压和/或降低的接地电源电压来实现升高的电源电压。向可编程逻辑器件架构中引入具有不同电压电平的电源线会增加复杂性。为了避免增加不必要的复杂性,可能期望限制对不同电源电压的使用。用于限制电源电平数量的一种合适方法是,使用用来向核心逻辑 18 和数据加载和读取电路 96 供电的同一接地电压  $V_{ss}$  来向阵列 88 供电。作为示例对这种类型的布置进行描述。然而,如果需要,可以使用其他布置,如其中针对阵列 88 的接地电压与针对核心逻辑 18 的接地电压不同的布置。

[0071] 使用线 86 和 90 来将  $V_{cchg}$  和  $V_{ss}$  分配给存储器单元阵列 88 中的每一个存储器单元 82。在图 6 的示例中,在阵列 88 中存在 3 行和 3 列的存储器单元 82。这仅仅是示例性的。诸如集成电路 10 的集成电路可以具有任何合适数量的存储器单元 82。作为示例,典型的存储器阵列可以具有按几百或几千行和列布置的几千或几百万个存储器单元 82。

[0072] 信号  $V_{ss}$  是恒定的。信号  $V_{cchg}$  可以是恒定值  $V_{cchg-high}$ 。例如, $V_{cchg-high}$  可以是诸如 1.6 伏特的恒定值。 $V_{cchg-high}$  的值大于  $V_{cc}$ (例如,1.2 伏特)。如果需要,功率调节器电路 84 可以生成时变电源信号以便于进行数据加载操作。例如,在数据加载操作过程中功率调节器电路 84 可以将  $V_{cchg}$  的值从  $V_{cchg-high}$  降低到  $V_{cchg-low}$  的值。在数据加载操作过程中降低  $V_{cchg}$  有助于改进针对存储器单元 82 的写入容限,并且/或者降低针对在对存储器单元进行加载时使用的地址晶体管的占用面积需求。这里作为示例对在阵列 88 中使用时变  $V_{cchg}$  进行描述。

[0073] 使用相对低的电源电平  $V_{cc}$  来向核心逻辑 18 供电,提供了诸如降低功耗的有益效果。用来向存储器单元 82 供电的电源电平比核心电源电平高,因为  $V_{cchg-high}-V_{ss}$  大于  $V_{cc}-V_{ss}$ 。然而,由于在阵列 88 中使用升高的电源电平而导致的阵列 88 的任何功耗效率损失都被性能的有益效果弥补了。

[0074] 数据加载和读取电路 96 对针对阵列 88 的清零和数据加载操作进行控制。数据加

载和读取电路 96 通过输入路径 98 从外部源接收配置数据。在典型的系统中,将配置数据从存储器和数据加载电路加载到可编程逻辑器件中。这种类型的电路(有时被称为配置器件)将配置数据加载到读写电路 115 内的寄存器中。地址译码器 102 可以通过路径 98 和输入 110 接收外部控制信号,或者可以在数据加载和读取电路 96 中内部生成寻址控制信号。

[0075] 数据加载和读取电路 96 在清零线 104(标为 CLR)上产生清零信号。通过电路 96 对信号 CLR 的宣称会将存储器阵列 88 的内容清零。通常在系统启动时或者在重新配置过程中执行清零操作。在对阵列进行了清零之后,去宣称 CLR 信号并加载配置数据。

[0076] 为了支持数据加载操作,可以通过路径 98 和 106 将配置数据加载到读写电路 115 内的寄存器中。这些寄存器将配置数据通过 DATA\_IN\_1 线、DATA\_IN\_2 线以及 DATA\_IN\_3 线 108 并行地施加给阵列 88。地址译码器 102 通过输入 110 从外部源或从数据加载和读取电路 96 中的电路接收寻址信息。地址译码器 102 系统地宣称和去宣称期望的地址线 112(即,ADD1、ADD2 或 ADD3)。当宣称了每一列中的地址线时,将数据线 108 上的数据加载到该列中的存储器单元 82 中。通过按此方式对每一列进行寻址,向整个阵列 88 加载了配置数据。

[0077] 在对阵列进行了加载之后,可以使用读写电路 115 的读取电路读出已编程到存储器单元 82 中的数据,来对正确的数据加载进行确认。利用该操作(有时被称为数据确认或数据读取)来确保在加载处理过程中没有出现差错。如果从阵列 88 读出的确认数据的值与在数据加载过程中使用的值不匹配,则可以生成差错消息,可以重复加载处理,或者可以采取其他合适的修正动作。

[0078] 在数据读取操作过程中,使用数据线 108 作为读取线。具体来说,使用 DATA\_IN\_1 线、DATA\_IN\_2 线以及 DATA\_IN\_3 线 108 将加载的配置数据值从被寻址的存储器单元 82 的列传送到读写电路 115。

[0079] 在完成了对数据加载处理的成功确认之后,利用所加载的配置数据来对器件 10 上的可编程逻辑的操作进行控制,使得可以在系统中使用器件 10。在正常操作过程中,各存储器单元 82 的输出 114 产生用于对可编程逻辑器件 10 的可编程核心逻辑 18 中的诸如 NMOS 传输晶体管的 NMOS 晶体管、诸如 PMOS 断电晶体管的 PMOS 晶体管或其他电路组件的栅极进行控制的对应静态控制信号。

[0080] 图 7 示出了例示性的存储器单元 82。在编程过程中,存储器单元 82 接收 DATA\_IN 线 108 上的数据。在已将数据加载到阵列中之后,执行数据确认操作以确定是否正确加载了数据。在数据确认操作过程中,DATA\_IN 线充当验证路径。当进行加载时,数据从图 6 的数据加载和读取电路 96 沿线 108 流到阵列 88 中。当执行数据确认操作时,数据从阵列 88 沿线 108 按相反方向流到图 6 的数据加载和读取电路 96 中。

[0081] 在加载操作和读取操作过程中,在地址线 112 上对地址信号 ADD 系统地进行宣称。这使得数据加载和读取电路 96(图 6)可以对阵列 88 的列进行寻址以进行数据写入或读取。

[0082] 图 7 的存储器单元 82 具有两个交叉耦合反相器。反相器 116 具有晶体管 120 和 122。反相器 118 具有晶体管 124 和 126。晶体管 120 和 124 是 p 沟道金属氧化物半导体晶体管。晶体管 122 和 126 是 n 沟道金属氧化物半导体晶体管。节点 N1 与 N2 上的电压具有相反的极性。当 N1 是低电平时, N2 是高电平,从而认为存储器单元 82 将被编程。在此情况下被编程存储器单元的内容是逻辑 1。当 N1 是高电平而 N2 是低电平时,该存储器单元包

含逻辑零,从而认为该存储器单元将被清零。

[0083] 如图 7 所示,节点 N2 上的电压与 DATA\_OUT 电压相同。通过诸如线 130 (如图 6 中的输出 114 所示) 的线,将 DATA\_OUT 信号施加给诸如晶体管 136 的可编程核心逻辑组件。

[0084] 根据本发明,线 112 上的地址信号 ADD 的电压在写入操作过程中具有相对高的电压,而在数据确认(读取)操作过程中具有相对低的电压。ADD 信号是由数据加载和读取电路 96 (图 6) 的地址译码器 102 生成的。根据一个合适的布置,地址信号在数据加载过程中从低电平值  $V_{ss}$  变化到高电平值  $V_{cchg-high}$ ,而在数据读取操作(数据确认)过程中从低电平值  $V_{ss}$  变化到高电平值  $V_{cchg-low}$  (例如,  $V_{cc}$ )。

[0085] 在写入操作过程中使用的升高的地址电压(在本示例中为 1.6 伏特)会比其它情况下更充分地使地址晶体管 132 导通,这会减小针对地址晶体管 132 的占用面积需求,并且/或者增大针对存储器单元 82 的写入容限。在数据确认读取操作过程中使用的降低的地址电压(在本示例中为 1.2 伏特)会增大存储器单元 82 的读取容限(read margin)。

[0086] 图 8 示出了可以在图 6 的数据加载和读取电路 96 中使用的例示性的读写电路 115。如图 8 所示,读写电路 115 包含寄存器 117,使用诸如路径 125 的路径将寄存器 117 连接成串。在数据加载操作过程中,使能寄存器 117 的输出 119,禁用读出放大器 121。在将数据移位到寄存器 117 中之后,各寄存器在对应的输出 119 上提供配置数据输出信号。每个输出 119 都连接到用于将配置数据加载到阵列 88 (图 6) 的存储器单元 82 中的相应 DATA\_IN 线 108。为了确认已正确地加载了配置数据,使用 DATA\_IN 线 108 执行数据读取操作。在数据读取操作过程中,禁用寄存器输出 119,而使能读出放大器 121。在本结构中,数据从阵列 88 的被寻址列通过 DATA\_IN 线 108、读出放大器 121 以及输入 123 流到寄存器 117 中。

[0087] 图 9 和 10 示出了在写入操作过程中可以使用的例示性电压。在图 9 和 10 的示例中,在正宣称地址信号的时段中将针对阵列中的存储器单元 82 的正电源电压  $V_{cchg}$  从高电平值  $V_{cchg-high}$  (例如,1.6 伏特)降低到低电平值  $V_{cchg-low}$  (例如,1.2 伏特)。这有助于改进存储器单元 82 的性能。图 9 示出了在将逻辑 1 写入存储器单元中时涉及的信号。图 10 示出了在将逻辑 0 写入存储器单元中时涉及的信号。

[0088] 成功的数据写入操作要求地址晶体管 132 比晶体管 124 和反相器 118 强,使得可以将来自线 108 的数据驱动到存储器单元 82 中。当降低  $V_{cchg}$  时,会削弱晶体管 124 的强度。结果,地址晶体管 132 不必与相反情况下要求的一样强。因此,降低  $V_{cchg}$  会增大针对存储器单元 82 的写入容限,并且/或者减小针对地址晶体管 132 的占用面积需求。根据设计考虑,降低的  $V_{cchg}$  值可以用来使得设计者减小各地址晶体管 132 消耗的占用面积(例如通过减小各地址晶体管 132 的栅极宽度),可以用来增大写入容限(通过利用地址晶体管 132 的增强强度来帮助确保在写入过程中不会出现差错),或者可以用来既减小占用面积又增大写入容限。

[0089] 当宣称了地址信号 ADD 时,其电压从 0 伏特 ( $V_{ss}$ ) 增大到升高的电平。在图 9 和 10 的示例中,该升高的电平是  $V_{cchg-high}$  (例如 1.6 伏特)。在写入操作过程中使用升高的地址信号,会增大在写入过程中施加给地址晶体管 132 的栅极 G 的电压的大小。在本示例中,向在写入操作中正被寻址的地址晶体管 132 的栅极施加的电压是  $V_{cchg-high}$  (1.6 伏特)。使用升高的值  $V_{cchg-high}$  来控制晶体管 132,会比使用较低的常规电压  $V_{cc}$  更充分地使晶体管 132 导通。因此,升高的地址信号值  $V_{cchg-high}$  起到了针对给定尺寸(栅极宽

度)有效地增大晶体管 132 的强度的作用。

[0090] 使用升高的地址信号电平而实现的地址晶体管 132 的有效强度的增大,有助于进一步增大存储器单元 82 的写入容限,并且/或者减小针对晶体管 132 的占用面积需求。

[0091] 在图 9 中,初始时将存储器单元 82 清零并向其加载逻辑 1。当宣称了地址信号 ADD 时,DATA\_OUT 线上的信号因而从低电平上升到高电平,如图 9 中的最下一条迹线所示。

[0092] 在按此方式对整个阵列 88 加载了数据之后,可以使电压 Vcchg 上升到它的高电平 Vcchg-high(例如 1.6 伏特)。这使得按比可编程核心逻辑正电源电压 Vcc(例如 1.2 伏特)高的电压将包含逻辑 1 的存储器单元 82 的输出提供给可编程核心逻辑,从而改进了性能。

[0093] 图 9 中的第一条迹线示出了由功率调节器电路 84(图 6)施加给线 86 以向阵列 88 中的存储器单元 82 供电的电源电压 Vcchg 在升高值 Vcchg-high 与降低值 Vcchg-low 之间变化。作为示例,Vcchg-low 可以等于 Vcc。将正电源电压 Vcchg 施加给反相器 116 和 118。Vcchg 的值在时间  $t_1$  时是高电平(在本示例中)。当期望将配置数据比特加载到存储器单元 82 中时,功率调节器电路 84 将 Vcchg 的值从 Vcchg-high 降低到 Vcchg-low(时间  $t_2$ )。在向阵列 88 的所有列加载了配置数据之后,功率调节器电路 84 将 Vcchg 的值升高到 Vcchg-high(在时间  $t_6$  处)。在  $t_6$  之后的时间,集成电路 10 正常操作,并且每个存储器单元 82 都产生值为 Vss(当存储了逻辑 0 时)或为 Vcchg-high(当存储了逻辑 1 时)的静态输出信号。

[0094] 如图 9 中的第二条迹线所示,本示例中的 Vcc 值保持恒定为 1.2 伏特。例如可以使用电源电压 Vcc 来向诸如可编程核心逻辑 18 和数据加载和读取电路 96(图 6)的电路供电。如果需要,可以在可编程逻辑器件 10 上使用其他电源电压。例如,可以使用更高的电压 Vccio 来向输入-输出电路 12(图 1)供电。为了避免不必要的复杂性,可以使用提供诸如 Vccio 的电压的电路来提供其他升高电压。例如,可以由同一电压源提供或从同一电压源获得 Vcchg-high 和 Vccio。

[0095] 如图 9 中的第三条迹线所示,在系统启动过程中执行了清零操作之后,线 104 上的清零信号 CLR 恒定为 0 伏特。

[0096] 图 9 中的第四条迹线示出了如何在时间  $t_4$  处宣称而在时间  $t_5$  处去宣称地址线 ADD。根据本发明,地址信号从低电平值 Vss 变化到升高值。在图 9 的示例中,升高地址信号具有 Vcchg-high 的大小。

[0097] 图 9 中的第五条迹线示出了在  $t_3$  之前的时间  $t$  处未限定线 108 上的 DATA\_IN 值。在时间  $t_3$  处,数据加载电路 96 产生低电平 DATA\_IN 信号(在图 9 的示例中)。

[0098] 当地址线 ADD 在  $t_4$  处取高电平时,晶体管 132(图 7)导通,这会将低电平 DATA\_IN 线 108 连接到节点 N1。ADD 的升高值有助于在时间  $t_4$  处强有力地使地址晶体管 132 导通。

[0099] 图 9 中的第六条迹线示出了节点 N1 上的电压。在时间  $t_1$  处,按 Vcchg-high(在本示例中)向反相器 116 和 118 供电,并且节点 N1 上的电压是 Vcchg-high。在时间  $t_2$  处,电压 Vcchg 从 Vcchg-high 下降到 Vcchg-low(迹线 1)。Vcchg 的该下降被反映在节点 N1 上的电压中,该电压也从 Vcchg-high 下降到 Vcchg-low。在时间  $t_4$  处,当由升高的 ADD 信号使晶体管 132 导通并将低电平 DATA\_IN 线连接到节点 N1 时,节点 N1 上的电压取低电平。

[0100] 由反相器 116 使节点 N1 上的低电压反相,使得节点 N2 上的电压和线 130 上的输

出电压 (DATA\_OUT) 在时间  $t_4$  处变高。这完成了对一系列存储器单元的编程,因此在时间  $t_5$  处去宣称了 ADD。

[0101] 当对每一列存储器单元进行加载时,  $V_{cchg}$  的电平通常保持为低电平。在对存储器单元 82 的所有期望的列加载了配置数据之后,功率调节器电路 84 使  $V_{cchg}$  从  $V_{cchg-low}$  上升到  $V_{cchg-high}$  (时间  $t_6$ )。

[0102] 根据施加给反相器 116 和 118 的电源电压  $V_{cchg-high}$ , 线 130 上的 DATA\_OUT 信号处于  $V_{cchg-high}$  的电压电平。升高了 DATA\_OUT 信号 (在本示例中,与  $V_{cc}$  和  $V_{cchg-low}$  的 1.2 伏特相比,  $V_{cchg-high}$  为 1.6 伏特), 因此, 向图 6 的可编程核心逻辑 18 中的诸如晶体管 136 (图 7) 的组件施加的静态高电平输出信号高得足以令人满意地对这些组件进行控制 (例如, 对于 n 沟道器件具有低导通电阻, 而对于 p 沟道器件具有低泄漏电流)。

[0103] 当在时间  $t_4$  处宣称地址信号 ADD 时使用的  $V_{cchg}$  的降低值 ( $V_{cchg-low}$ ) 和 ADD 的升高值 ( $V_{cchg-high}$ ) 放松了对 NMOS 晶体管 132 和 PMOS 晶体管 124 的相对强度的要求。如果尚未升高 ADD (并且如果尚未降低  $V_{cchg}$ ), 则可能要求更大的 NMOS 晶体管 132 和更大的伴随信号线, 来确保 PMOS 晶体管 124 可以胜任并且确保与数据加载操作相关联的 IR 下降不会过度。在数据加载过程中当升高了地址信号时并且当降低了电源电平时, 降低了针对 NMOS 晶体管 132 的强度要求和占用面积要求。

[0104] 图 10 中的迹线例示了将逻辑 0 加载到已清零的存储器单元 82 (图 7) 的过程。在此情况下, 存储器单元 82 包含逻辑零, 因此该加载过程不会改变其状态。

[0105] 如结合图 9 描述的那样, 图 10 中的第一条迹线示出了由功率调节器电路 84 (图 7) 施加给线 86 的电源电压  $V_{cchg}$  在升高值  $V_{cchg-high}$  与降低值  $V_{cchg-low}$  之间变化。  $V_{cchg}$  的值在时间  $t_1$  处是高电平。在时间  $t_2$  处, 为了准备进行配置数据加载, 功率调节器电路 84 将  $V_{cchg}$  的值从  $V_{cchg-high}$  降低到  $V_{cchg-low}$ 。时变电源电压  $V_{cchg}$  可以保持低电平, 直到对阵列 88 (图 6) 完全进行了加载为止。在向阵列 88 加载了配置数据之后, 功率调节器电路 84 使  $V_{cchg}$  的值升高到  $V_{cchg-high}$  (时间  $t_6$ )。

[0106] 如图 10 中的第二条迹线所示,  $V_{cc}$  的值保持恒定为 1.2 伏特 (在本示例中)。

[0107] 如图 10 中的第三条迹线所示, 在系统启动过程中执行了清零操作之后, 线 104 上的清零信号 CLR 恒定为 0 伏特。

[0108] 图 10 中的第四条迹线示出了如何在时间  $t_4$  处宣称而在时间  $t_5$  处去宣称地址线 ADD。地址信号的值从低电平值 0 伏特 ( $V_{ss}$ ) 变化到高电平值  $V_{cchg-high}$ 。使用该升高的电压有助于强有力地使地址晶体管 132 导通, 从而减小针对地址晶体管 132 的占用面积需求, 并且 / 或者增大针对存储器单元的写入容限。

[0109] 图 10 中的第五条迹线示出了在  $t_3$  之前的时间  $t$  处未限定线 108 上的 DATA\_IN 值。在时间  $t_3$  处, 数据加载电路 96 产生高电平 DATA\_IN 信号。

[0110] 当在时间  $t_4$  处宣称地址线 ADD 时, 地址信号上升到它的高电平值  $V_{cchg}$ 。这会使晶体管 132 (图 7) 导通, 并将高电平 DATA\_IN 线 108 连接到节点 N1。图 10 中的第六条迹线示出了节点 N1 上的电压。在时间  $t_3$  处, 存储器单元 82 处于其清零状态。在该状态下, 节点 N2 上的电压是接地电压。在时间  $t_3$  处按电压  $V_{cchg-low}$  向反相器 116 和 118 供电, 因此在节点 N1 处反相后的 N2 电压是  $V_{cchg-low}$ 。在时间  $t_4$  处, 当使地址线为  $V_{cchg-high}$  时, 使晶体管 132 导通, 并将高电平 DATA\_IN 线连接到节点 N1。节点 N1 已经是高电平, 因此如图

10 中的第七条迹线所示,在时间  $t_4$  处,数据输出线 DATA\_OUT 的状态保持低电平而不改变其状态。

[0111] 通过在时间  $t_5$  处去宣称地址线 ADD 而完成了将逻辑 0 比特加载到存储器单元 82 中的数据加载。在按此方式对阵列 88 的所有列加载了配置数据之后,在时间  $t_6$  处功率调节器电路 84 使 Vcchg 从 Vcchg-low 上升到 Vcchg-high。

[0112] 在现场中使用器件 10 之前,通常期望验证已经正确地将配置数据加载到阵列中了。为了确认已经正确地加载了数据,通过线 108 从该阵列读出数据。如果按此方式读出的数据与所加载的数据相同,则已正确地对该器件进行了编程,从而可以在系统中使用该器件。如果读出的数据与预期的配置数据不同,则出现了差错。

[0113] 图 11 和 12 示出了当确认正确地加载了配置数据时可能执行的数据读取操作。图 11 示出了在读取逻辑 1 时涉及的操作。图 12 示出了在读取逻辑 0 时涉及的操作。

[0114] 在读取操作过程中,系统地将地址信号 ADD 取高电平以对阵列 88 的列进行寻址。在读取操作过程中使用的地址信号 ADD 优选地相对较低(即,ADD 在读取操作过程中比在写入操作过程中小),以改进存储器单元的读取容限。

[0115] 在图 11 中,存储器单元包含逻辑 1(即,N1 是 0 并且认为该存储器单元是已编程的)。

[0116] 如图 11 中的第一条(最上一条)迹线所示,为了准备进行数据读取操作,信号 DATA\_IN 的电压初始时为高电平电压 Vcc(例如 1.2 伏特)。在  $t_1$  之前的时间,DATA\_IN 信号无效。在时间  $t_1$  处,图 6 的数据加载和读取电路 96 强烈地使 DATA\_IN 为高电平。在时间  $t_2$  处,数据加载和读取电路 96 微弱地使 DATA\_IN 保持高电平。在时间  $t_3$  处,如图 11 中的第二条迹线所示,通过使 ADD 从 0 伏特变为 Vadd-low,宣称了地址信号 ADD。作为示例,Vadd-low 的值可以是 1.2 伏特的电压(Vcc)。在数据读取操作过程中,如图 11 中的第三条迹线所示,可以使用升高的正电源电压 Vcchg-high(例如 1.6 伏特)向存储器单元 82 供电。

[0117] 在时间  $t_3$  处使 ADD 为高电平,会使晶体管 132(图 7)导通。在  $t_3$  与  $t_5$  之间的时间,晶体管 132 保持导通,因此将在 DATA\_IN 线 108 上微弱地保持的高电平电压暴露给反相器 118 的输出。节点 N2 处于高电平,因此晶体管 126 导通,并且反相器 118 的输出将低电平电压驱动到节点 N1 上。晶体管 126(图 7)比 DATA\_IN 线 108 上微弱地保持的“1”更强,因此 DATA\_IN 信号的值下降到约 0 伏特,如图 11 中的第一条迹线所示。在时间  $t_4$  处,数据加载和读取电路 96(图 1)检测到 DATA\_IN 信号的低电平电压,从而可以推断出 N1 上的电压低并且该存储器单元包含逻辑 1(在本示例中)。

[0118] 如图 11 中的第四条迹线所示,当在时间  $t_3$  处由 ADD 信号使地址晶体管 132 导通时,节点 N1 上的电压立即升高。图 11 中的第四条迹线所示的 N1 处的上升是不期望的。假如该上升太大,则反相器 116 会将“0”驱动到节点 N2 上,从而使存储器单元 82 的状态翻转。由于 ADD 信号的电压(即,在本示例中的 1.2 伏特的信号 Vadd-low)相对较低,所以晶体管 132 不会太强烈地导通。结果,很好地控制住了节点 N1 处的不期望上升的峰值并且不会出现翻转。

[0119] 在图 12 中,存储器单元包含逻辑 0(即,N1 处于高电平,N2 处于低电平,并且认为存储器单元已被清零)。

[0120] 在  $t_1$  之前的时间,如图 12 中的第一条(最上一条)迹线所示,DATA\_IN 信号是无

效的。在时间  $t_1$  处,图 6 的数据加载和读取电路 96 强烈地使 DATA\_IN 保持高电平。在时间  $t_2$  处,数据加载和读取电路 96 微弱地使 DATA\_IN 保持高电平。在时间  $t_3$  处,如图 12 中的第二条迹线所示,通过使 ADD 从 0 伏特变为  $V_{add-low}$  (例如  $V_{cc}$ ) 来宣称地址信号 ADD。如图 12 中的第三条迹线所示,可以使用升高的正电源电压  $V_{cchg-high}$  (例如,1.6 伏特) 来向存储器单元 82 供电。在时间  $t_3$  处使 ADD 为高电平,会使晶体管 132 (图 7) 导通。

[0121] 当在  $t_3$  与  $t_5$  之间的时间使 n 沟道金属氧化物半导体地址晶体管 132 导通时,允许信号从节点 N1 流到节点 133 (图 7)。由于将  $V_{cchg}$  升高为  $V_{cchg-high}$ ,所以在时间  $t_3$  处节点 N1 的电压是  $V_{cchg-high}$ ,这比在 DATA\_IN 线 108 上微弱地保持的电压  $V_{cc}$  高。将地址晶体管 132 的栅极 G 保持为地址信号电压  $V_{add-low}(V_{cc})$ 。由于诸如晶体管 132 的金属氧化物半导体晶体管的特性,所以当节点 N1 和 133 处的电压比地址线 112 (栅极 G) 处的电压减去 NMOS 晶体管 132 的阈值电压  $V_t$  高时,晶体管 132 截止并隔断了节点 N1 与 133。在  $t_3$  之前的时间微弱地保持的 DATA\_IN 线 108 的电压电平因此在时间  $t_3$  处保持不变,使得在时间  $t_3$  处节点 N1 的电压不会发生变化。在时间  $t_4$  处,数据加载和读取电路 96 (图 6) 检测到 DATA\_IN 信号的高电平电压,从而可以推断出 N1 上的电压为高电平并且该存储器单元包含逻辑 0。如图 12 中的第二条迹线所示,在时间  $t_5$  处去宣称了地址信号。

[0122] 如图 12 中的第四条迹线所示,即使地址信号是高电平,在时间  $t_3$  到  $t_5$  期间节点 N1 上的电压也保持恒定为  $V_{cchg-high}$ 。这是因为将栅极电压 G 的上限限定为相对低的地址信号电压  $V_{add-low}(V_{cc})$ 。对栅极电压 G 施加的上限限制了由反相器 118 驱动到输出节点 133 中的最大电压,因此在读取操作过程中很好地控制住了电压 N1。

[0123] 图 13 示出了对可编程逻辑器件集成电路 10 中的存储器单元 82 进行操作时涉及的步骤的流程图。

[0124] 在步骤 138 处,使用清零线 104 对存储器单元进行清零,以准备进行数据加载。

[0125] 在步骤 140 处,可以使用诸如  $V_{cchg-low}$  (例如,1.2 伏特) 的电源电压向存储器单元 82 供电。如图 9 和 10 中的第一条迹线所示,该值可能是从已有的相对较高的值  $V_{cchg-high}$  降低得来的,以准备进行配置数据加载操作。另选的是,可以使用正电源电压  $V_{cchg-low}$  对阵列 88 上电。如果使用  $V_{cchg-low}$  对阵列 88 上电,则在数据加载操作过程中可以保持该相对低的正电源电压,以便于进行编程。

[0126] 在步骤 142 处,通过输入 106 (图 6) 将配置数据移位到读写电路 115 的寄存器 117 中。

[0127] 在步骤 144 处,在输入 110 处向图 6 的地址译码器 102 施加控制信号。该控制信号指示地址译码器 102 宣称期望的地址线 112,以对阵列 88 中的存储器单元列进行寻址。作为响应,宣称了阵列中的一条地址线 112,以对阵列 88 中的存储器单元 82 的对应列进行寻址。如结合图 9 和 10 所描述的,优选的是,使地址信号 ADD 从低电平值  $V_{ss}$  (例如,0 伏特) 变为升高值  $V_{cchg-high}$  (例如,1.6 伏特)。

[0128] 在步骤 144 期间使用的升高 ADD 信号电压的相对高的值 (例如  $V_{cchg-high}$ ) 和存储器单元阵列正电源电压的相对低的值 (例如  $V_{cchg-low}$ ),有助于减小针对地址晶体管 132 的占用面积需求,并且 / 或者增大针对存储器单元的写入容限。

[0129] 在步骤 146 处,在将针对被寻址的存储器单元列的配置数据通过数据线 108 加载到阵列 88 中之后,去宣称地址线 112 (例如,如结合图 9 和 10 所描述的,通过将地址线的电

压从其升高电平  $V_{cchg-high}$  降低到接地电压  $V_{ss}$  )。

[0130] 如线 148 所示,配置数据加载过程持续至,数据加载和读取电路 96 系统地宣称了阵列 88 中的所有地址线 112 并且对阵列 88 的所有存储器单元 82 进行了加载。

[0131] 一旦对阵列 88 完全地加载了配置数据,功率调节器电路 84 就可以将存储器单元正电源电压升高到  $V_{cchg-high}$ ,以准备进行数据确认操作。

[0132] 在步骤 150 处,数据加载和读取电路 96 对一条地址线 112 进行宣称。优选的是,使该地址线 112 的电压从  $V_{ss}$  上升到相对低的值  $V_{add-low}$  (例如,  $V_{cc}$ ),以帮助改进存储器单元 82 的读取容限。由数据加载和读取电路 96 将已加载到被寻址的存储器单元列中的数据从这些存储器单元中读出来,如结合图 11 和 12 所描述的 (步骤 152)。

[0133] 在步骤 154 处,通过将所宣称的地址线的电压从  $V_{add-low}$  降低到  $V_{ss}$ ,来对该地址线进行去宣称。

[0134] 如线 156 所示,数据读取操作持续至,数据加载和读取电路 96 从阵列 88 系统地读出了所有加载的配置数据。将已读出的数据与所加载的数据进行比较。如果在所加载的数据与所读出的数据之间存在差异,则可以采取修正动作。

[0135] 如果数据确认过程成功,则可以在系统的正常操作中使用该可编程逻辑器件集成电路 10 (步骤 158)。在正常操作过程中,功率调节器电路 84 按相对于用来向可编程核心逻辑 18 供电的电源电平而升高的电源电平向阵列 88 供电。例如,可以按  $V_{cchg-high}$  和  $V_{ss}$  向存储器单元 82 供电。当按此方式供电时,包含逻辑 0 的存储器单元 82 在它们的  $DATA\_OUT$  端子处按  $V_{ss}$  产生控制信号,而包含逻辑 1 的存储器单元 82 在它们的  $DATA\_OUT$  端子处按  $V_{cchg-high}$  产生控制信号。 $V_{cchg-high}$  的大小比用来向可编程核心逻辑 18 供电的电压  $V_{cc}$  大,这增强了性能。

[0136] 在结合图 13 描述的例示性布置中,由提供时变正电源电压  $V_{cchg}$  和接地电压  $V_{ss}$  的动态功率调节器电路来向存储器阵列 88 供电。在正常操作过程中和在数据读取操作过程中将电源电压  $V_{cchg}$  升高为  $V_{cchg-high}$ 。在数据写入操作过程中将电源电压  $V_{cchg}$  降低到  $V_{cchg-low}$ 。通常,  $V_{cchg-high}$  和  $V_{cchg-low}$  可以是任何合适的电压。 $V_{cchg-low}$  的一个合适电压是容易获得的电源电压  $V_{cc}$ 。然而,如果需要,  $V_{cchg-low}$  可以低于或高于  $V_{cc}$ 。 $V_{cchg-low}$  的值例如可以是 1.2 伏特,或者可以使用更低或更高的值。 $V_{cchg-high}$  可以是 1.6 伏特 (例如,当  $V_{cc}$  为 1.2 伏特时),可以在 1.2 伏特到 1.4 伏特的范围内,可以在 1.2 伏特到 1.6 伏特的范围内,可以高于 1.6 伏特,或者可以低于 1.2 伏特 (即,当  $V_{cc}$  低于 1.2 伏特时)。

[0137] 根据需要升高和降低  $V_{cchg}$  的值提供了诸如降低针对阵列中的地址晶体管的占用面积需求和增大写入容限的有益效果。然而,如果需要,可以使用固定值  $V_{cchg}$ 。例如,可以将  $V_{cchg}$  的值固定为  $V_{cc}$  或者可以固定为升高电平 (例如,当将  $V_{cc}$  固定为 1.2 伏特时将  $V_{cchg}$  的值固定为 1.6 伏特)。无论使用哪种类型的存储器单元电源电压方案,都可以获得在数据写入操作和数据读取操作过程中针对地址信号使用不同的最大电压电平的有益效果。

[0138] 正确的寻址操作要求地址信号落在特定范围内。在数据写入操作过程中使用升高的地址信号  $V_{cchg-high}$  和在数据读取操作过程中使用低电平信号  $V_{cc}$  仅仅是例示性的。

[0139] 通常,当使用双电平电源电压  $V_{cchg}$  ( $V_{cchg-high}/V_{cchg-low}$ ) 时,升高的地址信



号电压  $V_{cchg-high}$  是有利的,因为在集成电路 10 上已经可以提供  $V_{cchg-high}$ 。如果需要,可以使用比  $V_{cchg-high}$  低或高的高地址电压  $V_{add-high}$ 。类似地,当  $V_{cchg-low}$  的值和数据线信号为  $V_{cc}$  时,使用数据读取地址信号电压  $V_{cc}$  是有利的。如果需要,  $V_{add-low}$  可以低于或高于  $V_{cc}$ 。  $V_{add-low}$  通常低于  $V_{add-high}$ ,从而可以同时改进写入容限(和/或地址晶体管占用面积需求)和读取容限。

[0140] 使用正电源电压  $V_{cchg-high}$  和接地电压  $V_{ss}$  来升高存储器单元电源电压通常是令人满意的。然而,如果需要,可以使用其他布置由功率调节器电路 84 提供升高的电源电压。例如,可以使用处于  $V_{cc}$  的正电源电压和比  $V_{ss}$  低的接地电源电压来提供升高的电源电平。根据该类型的布置,由于存储器单元正电源电压与存储器单元接地电压之差大于  $V_{cc}$  与  $V_{ss}$ (用来向可编程核心逻辑 18 的电路供电的电源电平)之差,所以升高了存储器单元电源电压。如果需要,可以调节在集成电路 10 中使用的金属氧化物晶体管(例如,存储器单元晶体管)的阈值电压,以适应改变的正电源电压和接地电源电压。

[0141] 当针对存储器阵列使用固定的单电平电源电压  $V_{cchg}$  时,升高的地址信号  $V_{add-high}$  可以比  $V_{cchg}$  高或者比  $V_{cchg}$  低。如果将存储器单元电源电压固定为  $V_{cc}$ ,则在数据写入操作过程中使用的地址信号  $V_{add-high}$  可以比  $V_{cc}$  高。地址信号  $V_{add-low}$  通常比  $V_{add-high}$  低。  $V_{add-low}$  电平  $V_{cc}$  通常是合适的,因为在集成电路上容易获得电压  $V_{cc}$ 。然而,如果需要,  $V_{add-low}$  可以比  $V_{cc}$  低或高。

[0142] 针对 DATA\_IN 线 108 上的数据信号使用的电压通常是  $V_{cc}$ ,因为这是核心逻辑 18 中的逻辑信号使用的电压。如果需要,该数据信号可以使用更高或更低的值。

[0143] 在存储器单元的反相器中存在与 n 沟道和 p 沟道金属氧化物半导体晶体管相关联的阈值电压  $V_{tn}$  和  $V_{tp}$ 。反相器的正确操作要求反相器电压超过  $V_{tn}$  与  $V_{tp}$  之和。有时将  $V_{tn}$  与  $V_{tp}$  之和称为  $V_{function}$ 。通常,为了使存储器单元正确地操作,  $V_{cchg-low}$  必须高于  $V_{function}$ 。优选的是,  $V_{add-low}$  的值高于  $V_{function}$  而低于  $V_{cchg-high}$ 。

[0144] 以上说明仅仅是对本发明原理的例示,而本领域的技术人员在不脱离本发明的范围和精神的情况下可以进行各种修改。

[0145] 本申请要求 2005 年 11 月 17 日提交的美国临时专利申请第 60/737,868 号的优先权,通过引用将其全部内容并入于此。

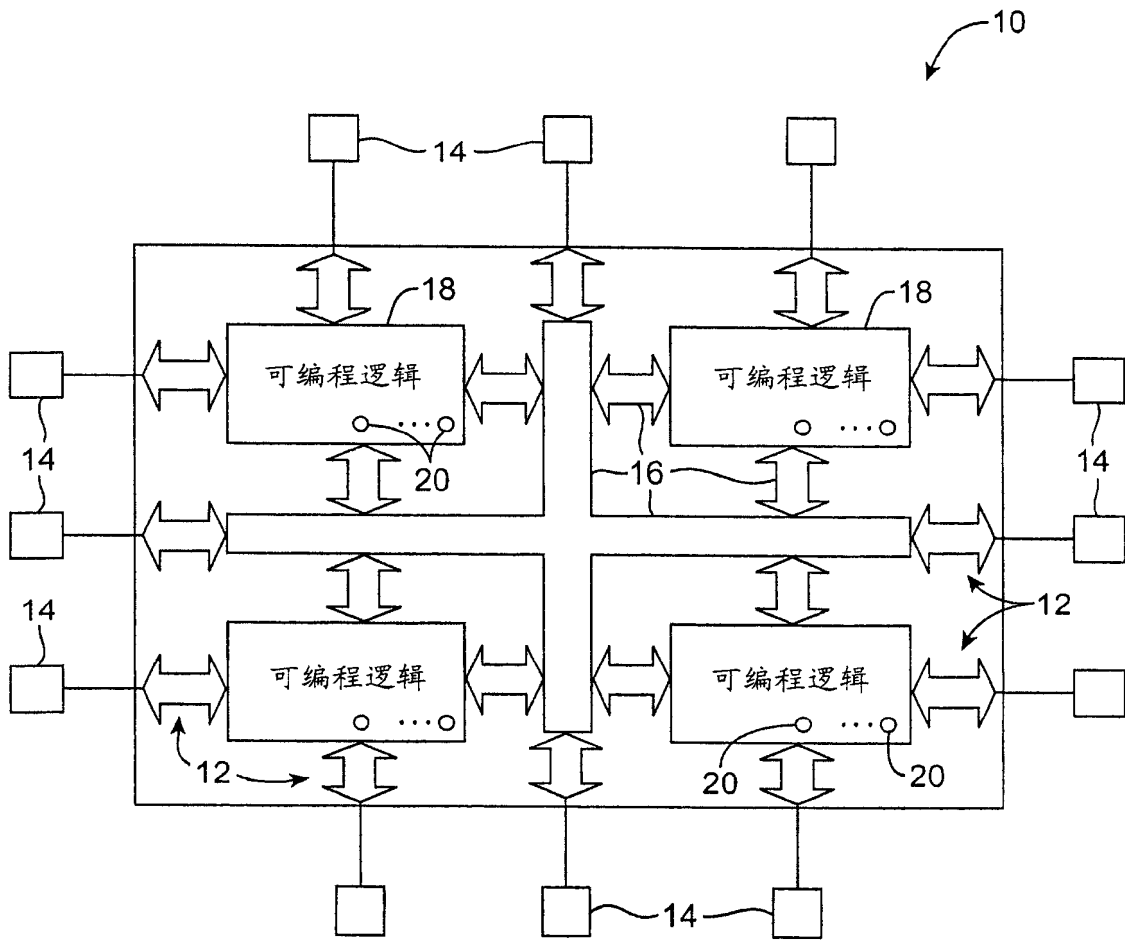


图 1

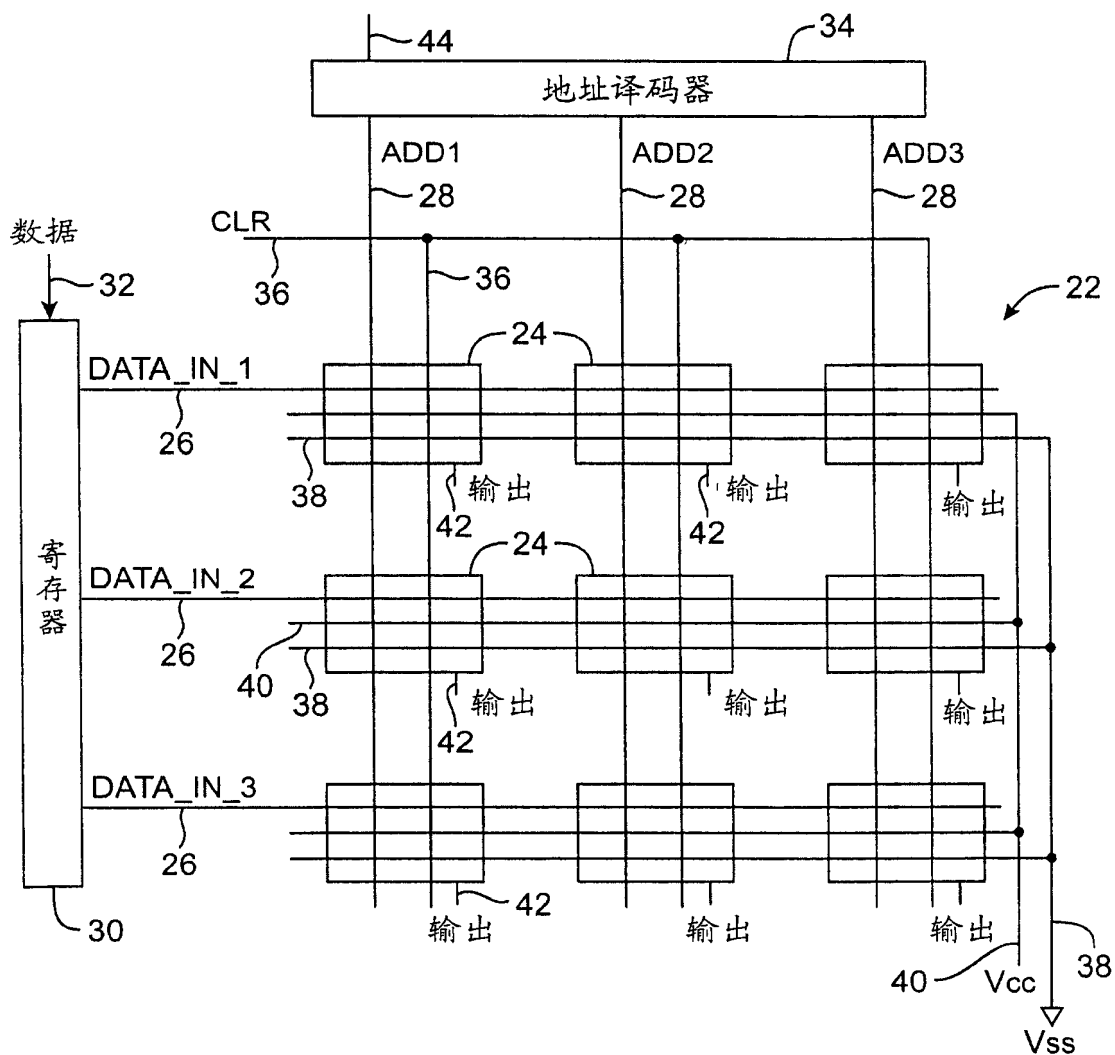


图 2 现有技术

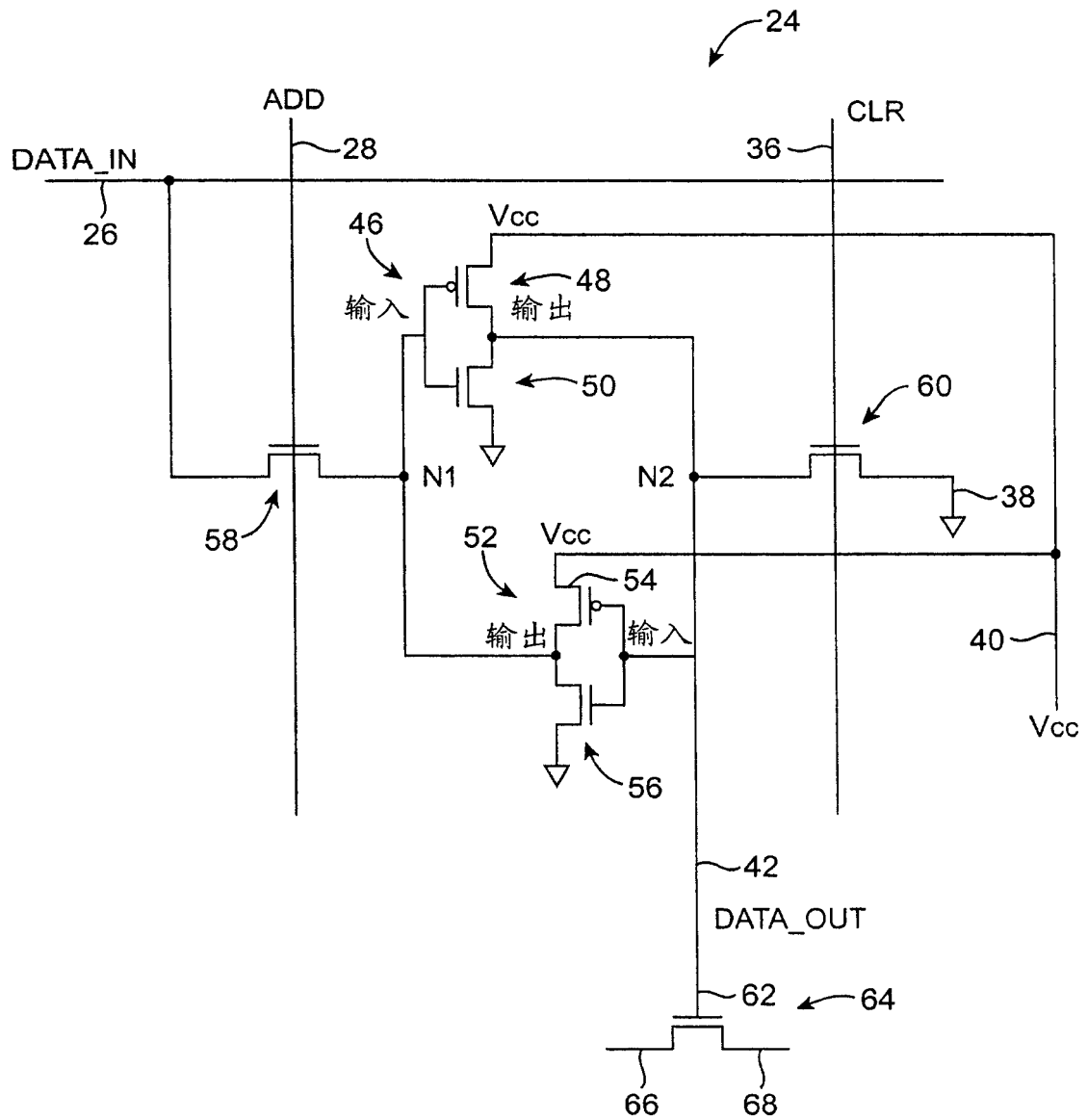


图3 现有技术

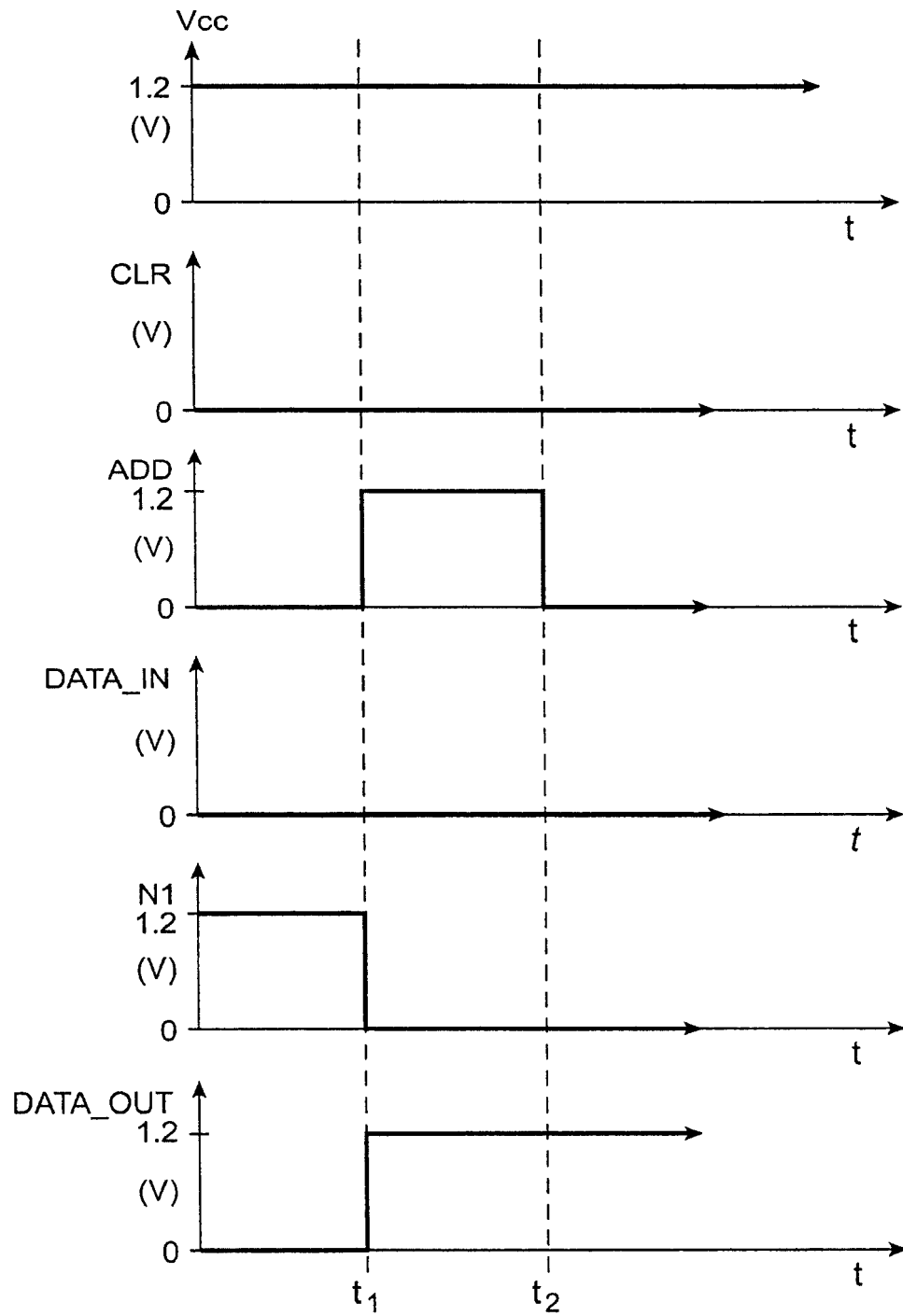


图4 现有技术

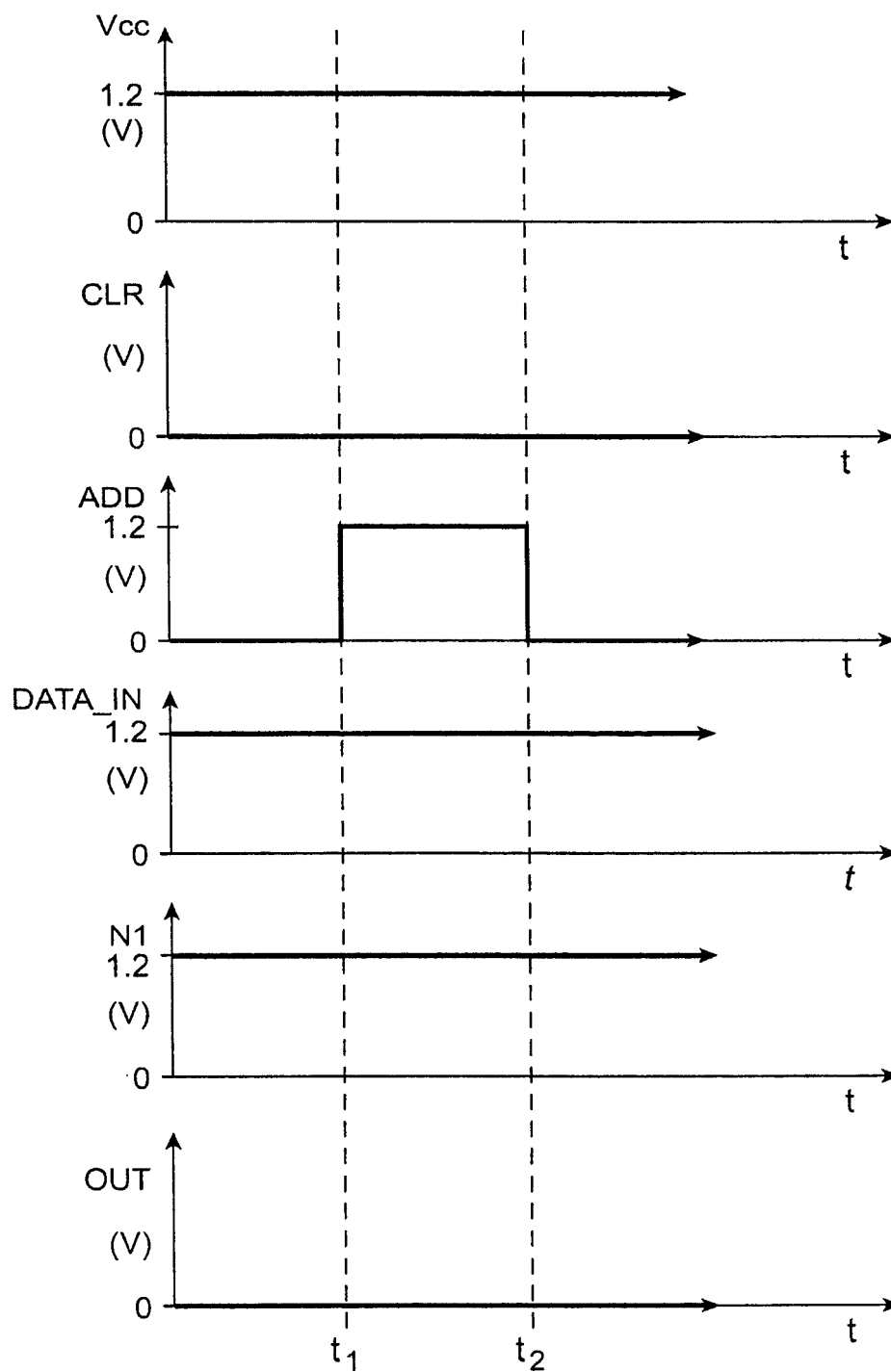


图5 现有技术

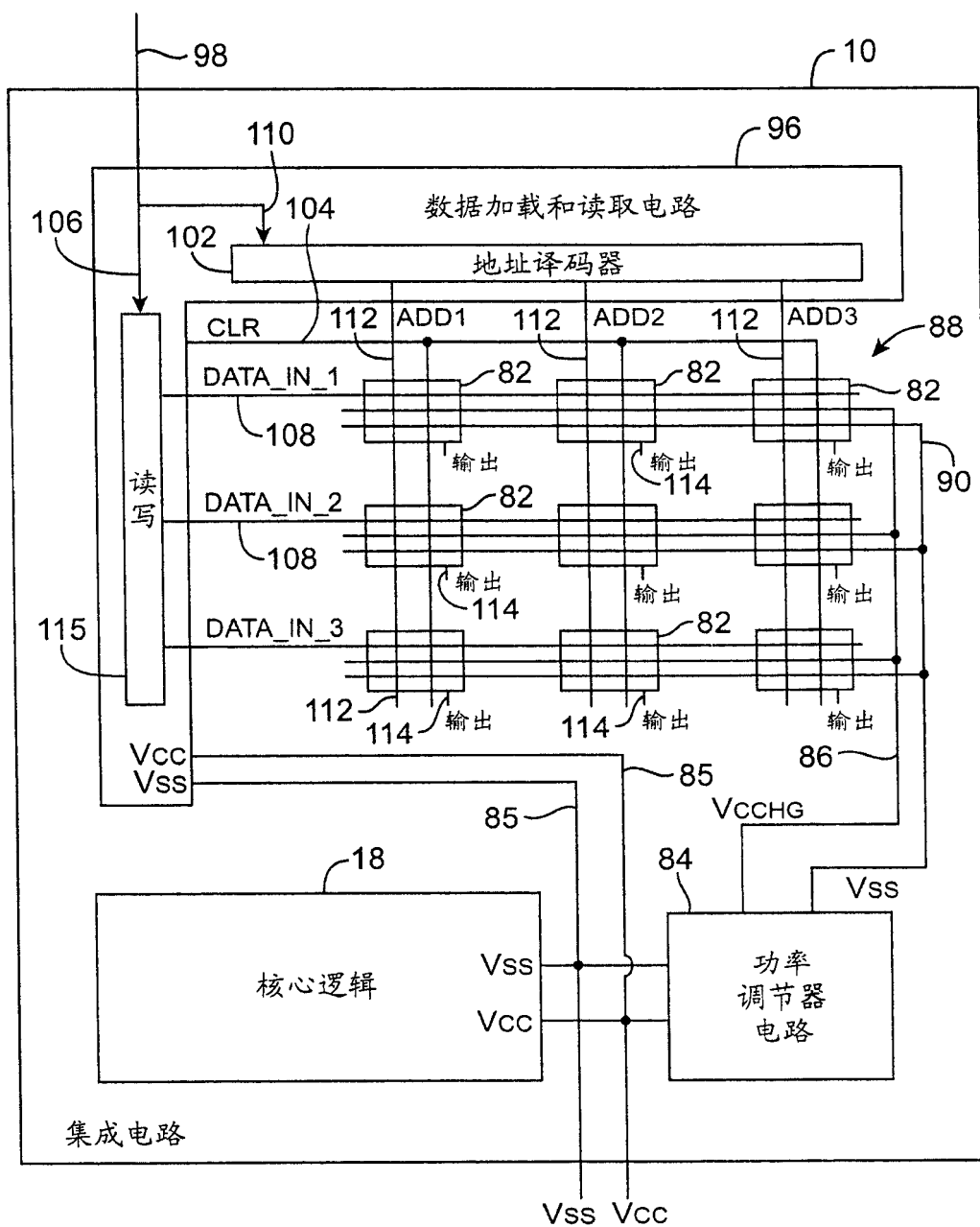


图 6

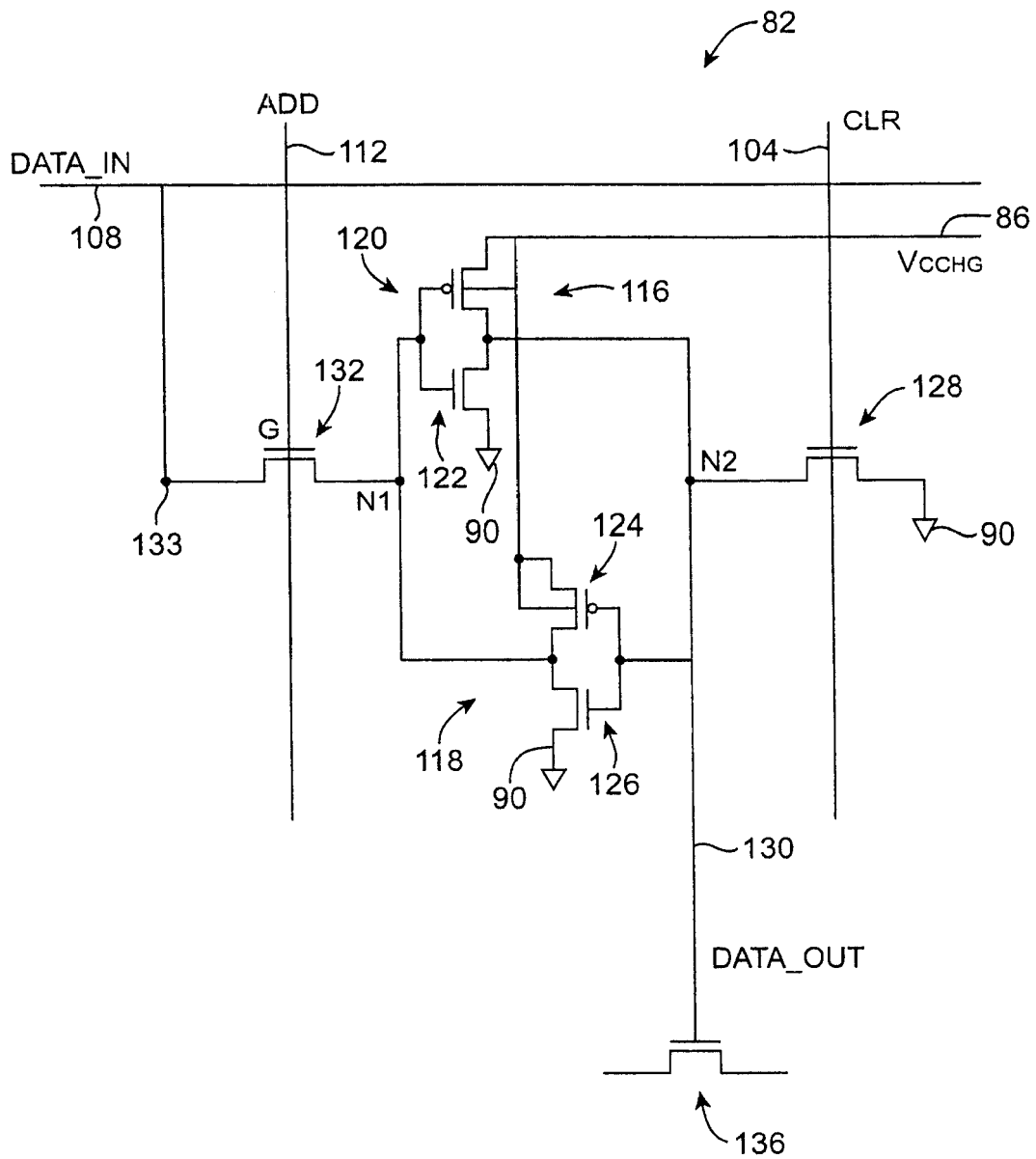


图 7



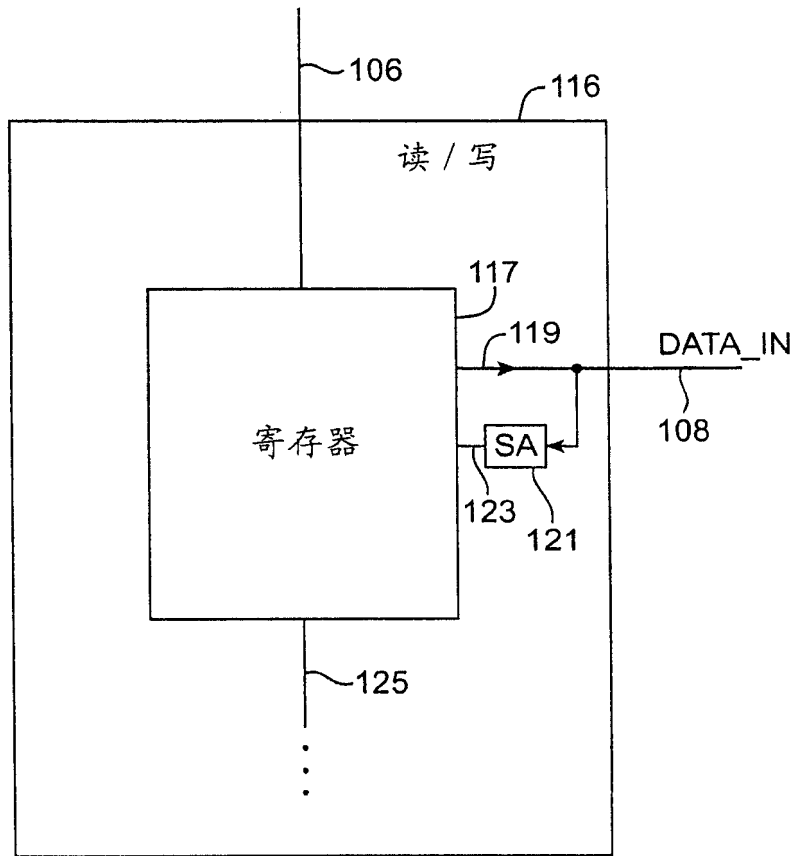


图 8

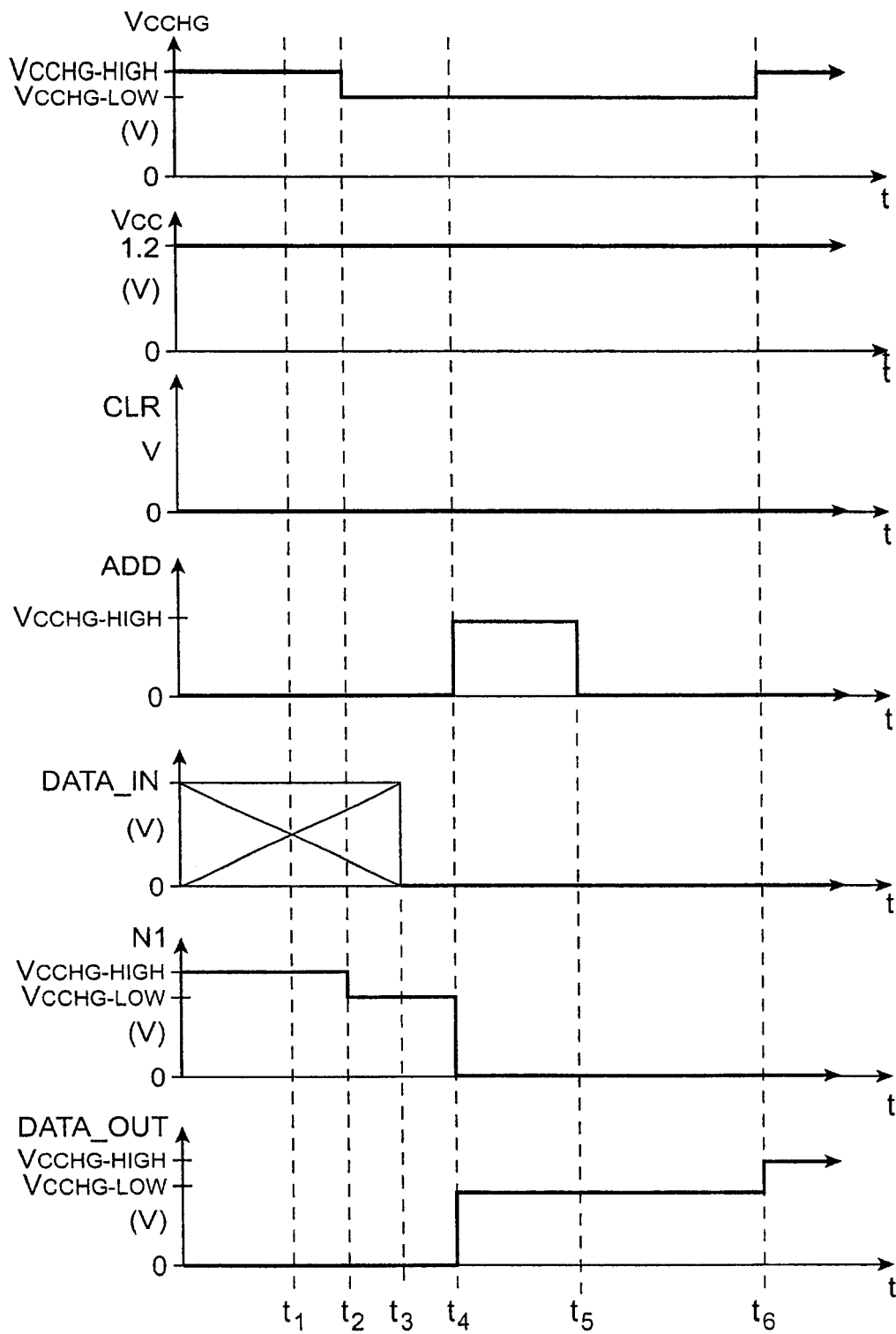


图 9

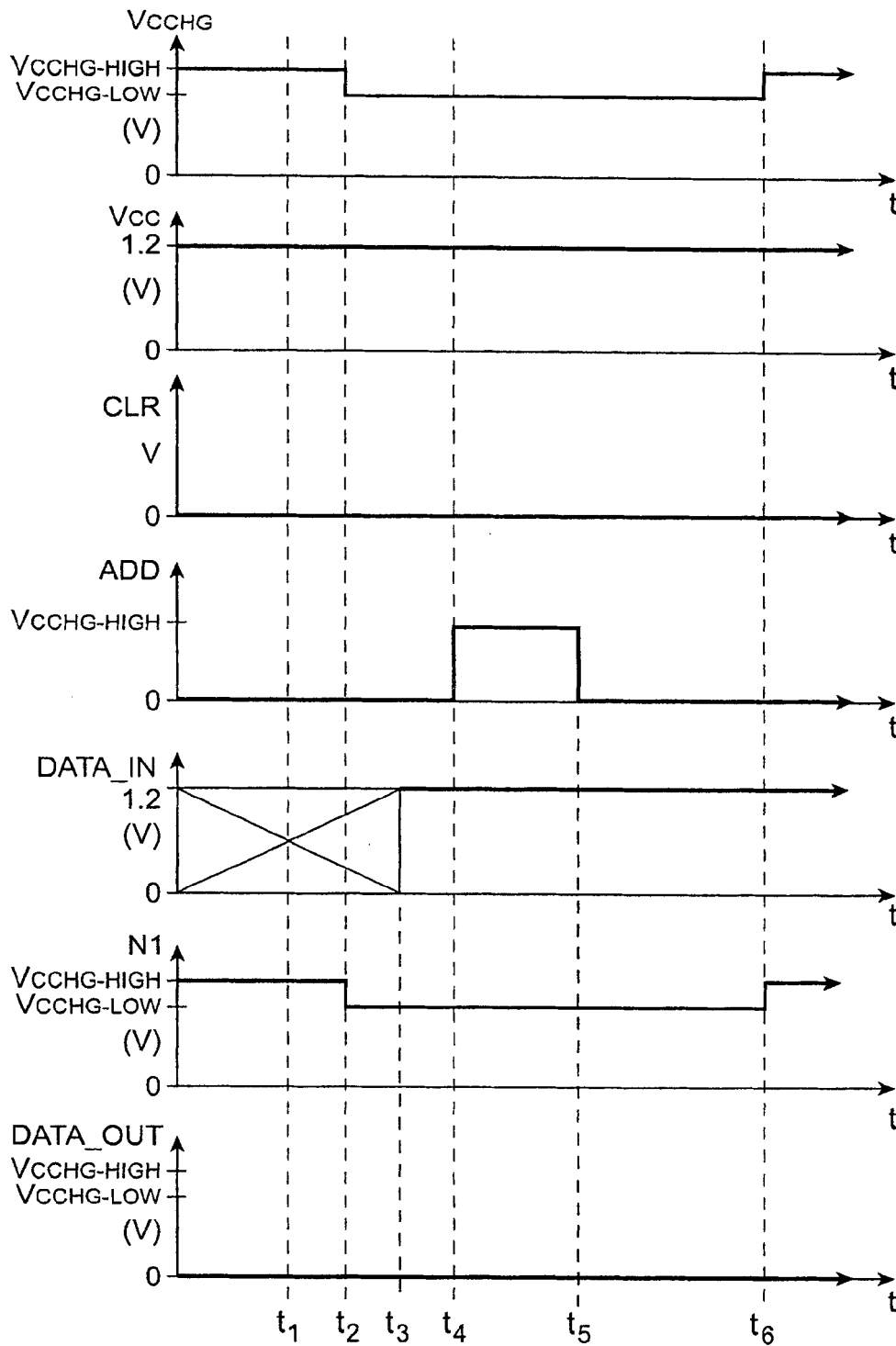


图 10

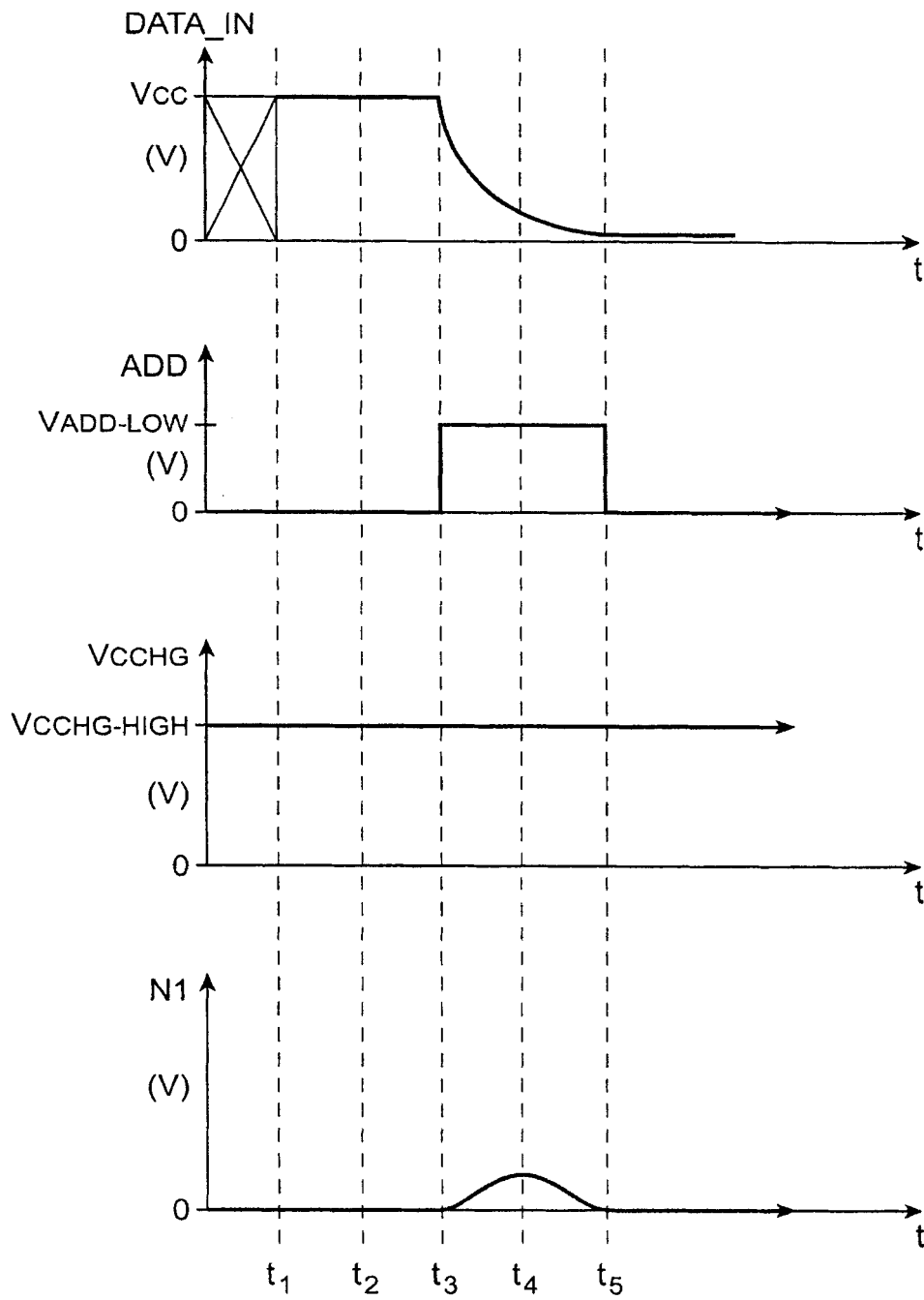


图 11

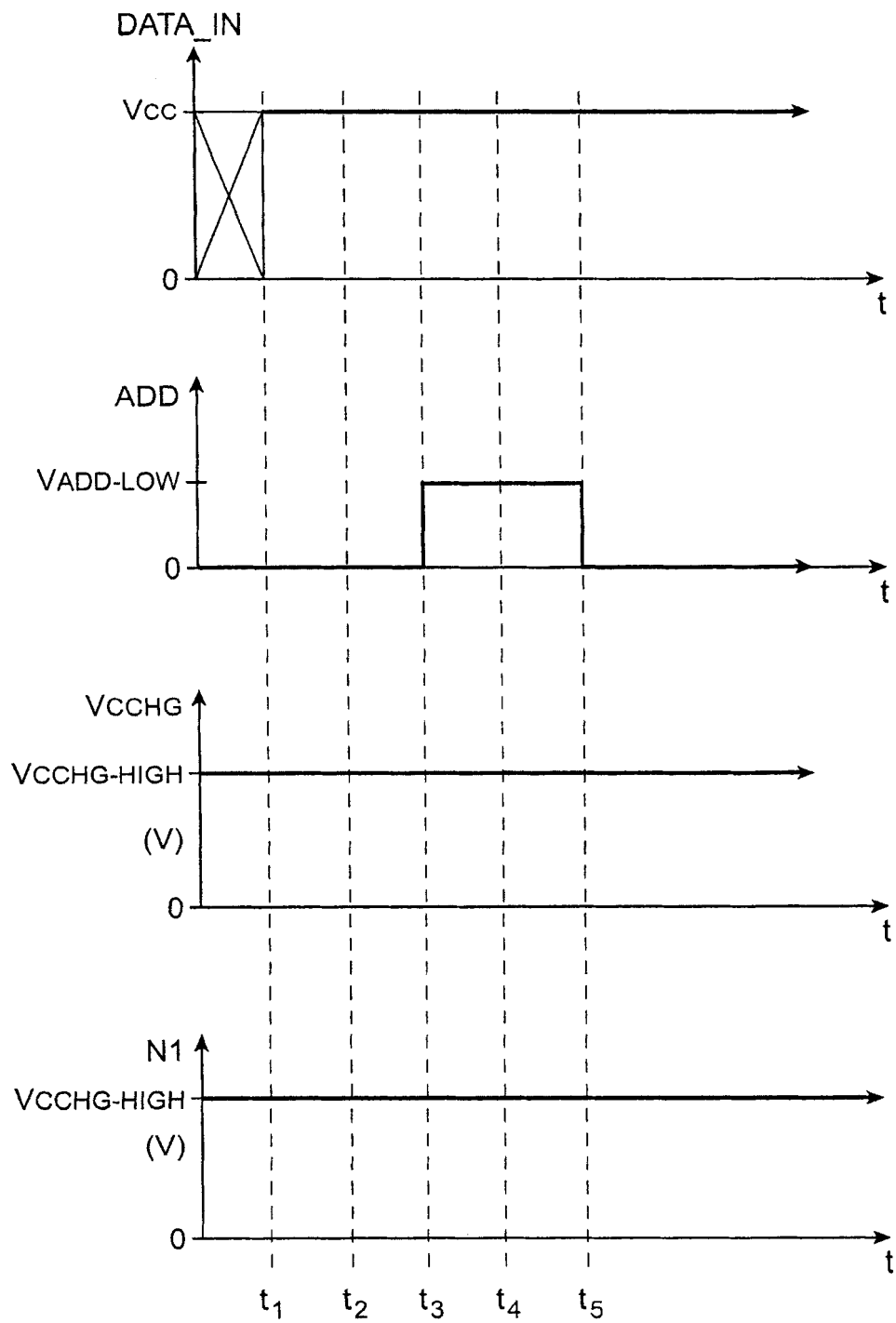


图 12

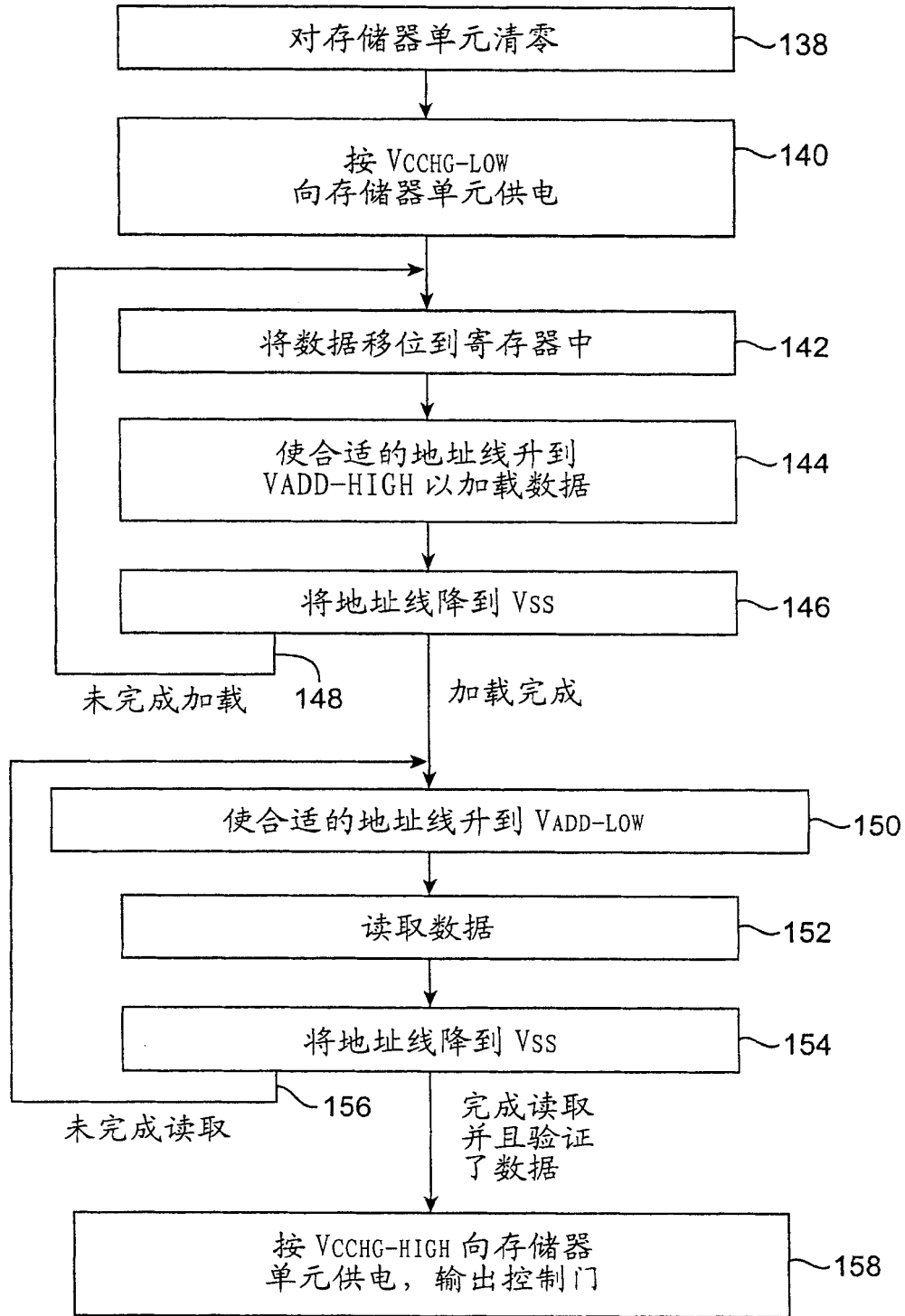


图 13