

基于 FPGA 的嵌入式块 SRAM 的设计

1 引言

对于逻辑芯片的嵌入存储器来说，嵌入式 SRAM 是最常用的一种，其典型的应用包括片上缓冲器、高速缓冲存储器、寄存器堆等。除非用到某些特殊的结构，标准的六管单元 (6T) SRAM 对于逻辑工艺有着很好的兼容性。对于小于 2Mb 存储器的应用，嵌入式 SRAM 可能有更好的成本效率并通常首先考虑。

Xilinx 公司 SRAM 型 FPGA 主要由配置存储器、布线资源、可编程 I/O、可编程逻辑单元 CLB、块存储器 BRAM 和数字时钟管理模块组成。它包含了分布式 RAM，位于 CLB 中。每个 CLB 包含了 $16 \times 1\text{bit}$ 的 SRAM 结构。BRAM 的加入既增加了 RAM 的容量，也可构成大型 LUT，更完善了 CLB 的功能。

2 BRAM 块划分

现代数字系统对存储器容量的存储速率要求越来越高，读访问时间就是一个重要参数，它是从地址信号的出现到存储在该地址上的数据在输出端出现的时间延迟。提高 BRAM 读取速度的一个有效办法是减小位线和字线上的总负载电容，这可以通过减少连接在同一字线和位线上的存储单元数目来实现，即采用存储阵列分块技术。本电路采用设计多个 BRAM 的方法，每个 BRAM 都有自己的译码电路、敏感放大器和数据通道，各个 BRAM 独立工作，每个 BRAM 的读取时间得到了大大提高。

3 BRAM 块设计

3.1 BRAM 与布线资源接口

FPGA 中每个 BRAM 块都嵌在内部连线结构中，与 BRAM 直接相连的有 RAMLINE、VLONG 和 GLOBAL。左边 32 根 RAMLINE 提供 BRAM 的地址输入，也可以提供控制信号 (CLK、WE、ENA、RST) 的输入。左边两组 16 根 RAMLINE 一起布线提供 BRAM 双端口的数据输入，右边两组 RAMLINE 提供 BRAM 双端口数据输出通道。4 根 GLOBLE 全局时钟线优化用作时钟输入，提供较短的延迟和最小的失真。VLONG 也被专门用作 BRAM 中 WE、ENA、RST 的控制输入。RAMLINE 为 BRAM 专有布线，如从水平方向的 SINGLE、UNIHEX、BIHEX 通过可编程开关矩阵 PSM 把信号输送到 RAMLINE 上，进而送到 BRAM 用作地址、数据。而 BRAM 的输出也通过 RAMLINE 最终送到 HLONG 上。

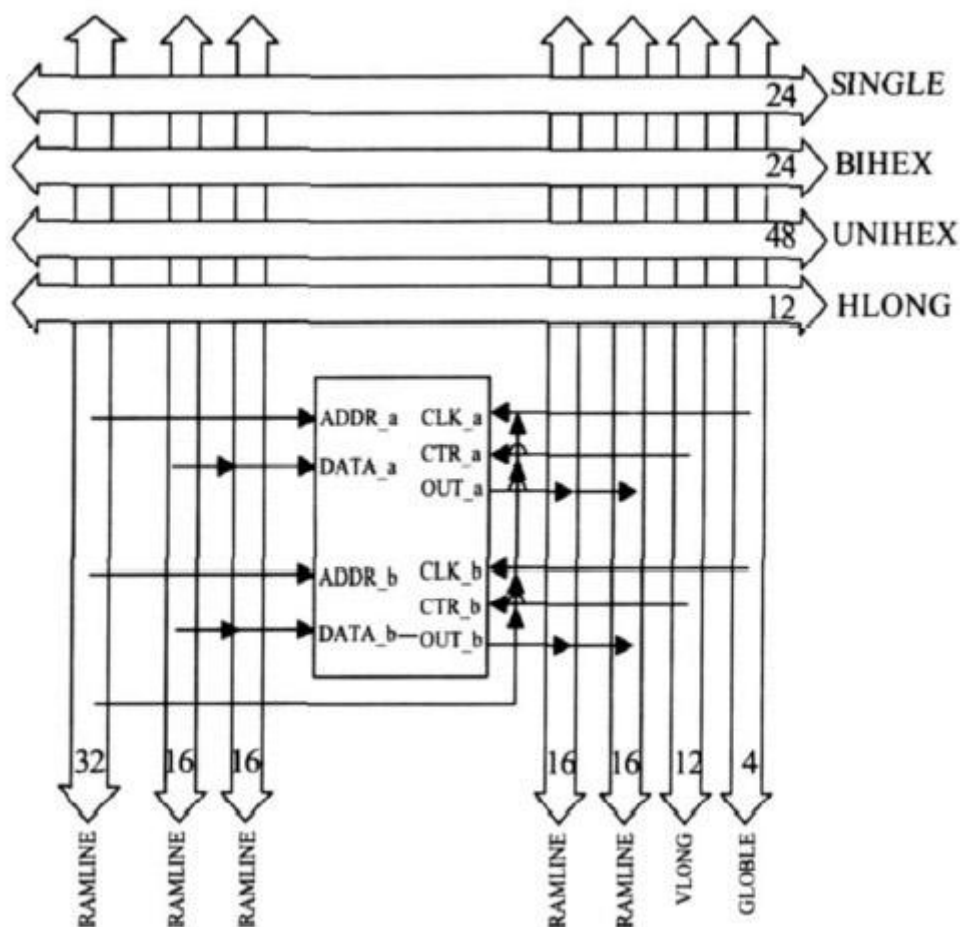


图 1 BRAM 周围布线

相邻 BRAM 的 RAMLINE 也可通过三态门连到下一级的 RAMLINE，于是整列中的 BRAM 可共享 RAMLINE 上的数据。每个 BRAM 与 FPGA 其他电路的相连主要通过水平方向的 4 组主要互连线完成。

3.2 BRAM 内部设计

BRAM 为真正的双端口 RAM，两个端口完全独立，每个端口可以配置为读写端口，并可以把 BRAM 配置成特定的数据宽度。

3.2.1 可配置数据位宽实现方法

配置逻辑中三位控制信号 WIDTH_SEL 《0：2》连到 BRAM 中，同时对地址宽度、数据宽度进行控制。

由于 BRAM 可以实现 1、2、4、8、16 位的任意位宽，所以地址总线宽度、数据总线宽度都必须满足其中任意一种模式下的要求。于是设计时使地址总线宽度为各种模式下的最大值，即 1 位时的地址宽度 《11：0》，其他模式下可使不用的地址位使能无效，进而获得所需的地址位。数据总线宽度也设置为各种情况

下的最大值，即 16 位时的数据宽度《15：0》，其他情况下选择有用的数据位进行存储。

表 1 可见 WIDTH_SEL《0：2》对地址使能的控制，主要在于对地址《11：8》的控制，其他位地址《7：0》则一直有效。

表 1 不同数据位宽的地址使能

工作模式	Width_sel	Addr<11>_en	Addr<10>_en	Addr<9>_cn	Addr<8>_cn
S1	7	1	1	1	1
S2	6	0	1	1	1
S4	5	0	0	1	1
S8	4	0	0	0	1
S16	3	0	0	0	0

由 WIDTH_SEL《0：2》另外译码产生一组数据控制信号，分别为 S_1、S_2、S_4、S_8、S_16 控制数据如何分配到位线上。这当中* 根位线实行了分片，每片 4 根：

S_1 有效：DI《0》可分配到 16 片中的任何一片上。

S_2 有效：DI《0：1》可分配到《0：1》、《2：3》、《4：5》？任何相邻两片上，每片 1 位数据。

S_4 有效：DI《0：3》可分配到《0：3》、《4：7》、《8：11》、《12：15》任何相邻四片上，每片 1 位数据。

S_8 有效：DI《0：7》可分配到《0：7》或《8：15》8 片上，每片 1 位数据。

S_16 有效：DI《0：15》刚好分配到 16 片上，每片 1 位数据。

至于上述究竟存储到哪些片上以及具体存储到片内哪根位线上则由列译码控制。

3.2.2 译码控制

行译码采用了常用的 3-8 译码器，3-8 译码器内由与门组成。第一级用两个 3-8 译码器，输入端接入行地址 ADDR 《5：0》，第二级用 64 个与门把第一级译码进一步译出来，可实现 64 行中选出 1 行。

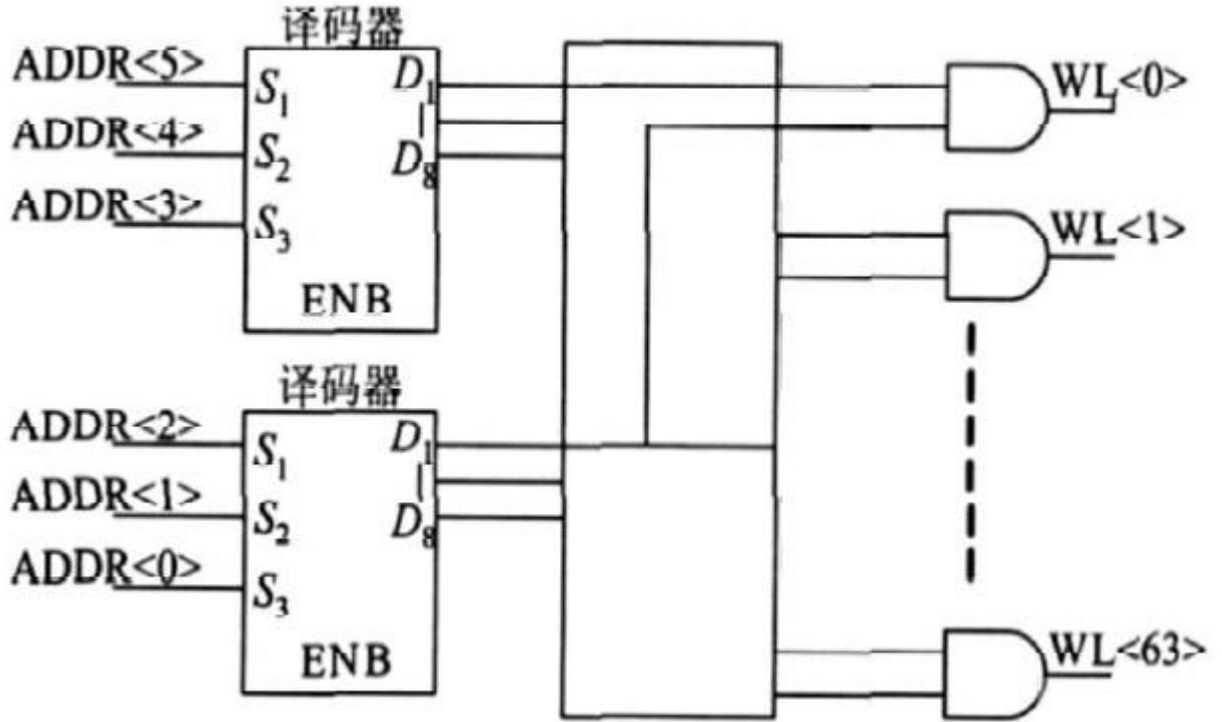


图 2 64 选 1 行译码

列译码相对较复杂，首先将列地址分为两组，一组用于片选译，一组用于片内译码。片选地址由 ADDR 《11：8》组成，片内译码由 ADDR 《7：6》组成。

片选地址译码由地址和地址使能组成，而地址使能则是由 WIDTH_SEL 《0：2》配置决定的。

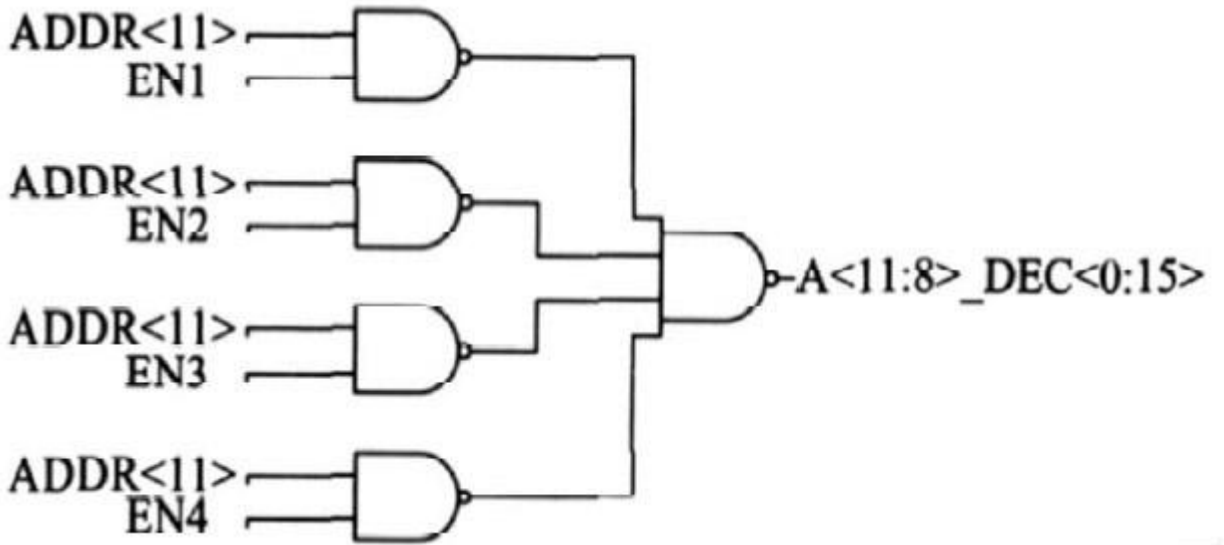


图3 片选译码

译码所得的 $A\langle 11:8\rangle_DEC\langle 0:15\rangle$ 即可实现片选存储。当配置为 1 位时，4 位地址均有效，译出的 16 位中只有 1 位有效，只能选择 16 片中的 1 片。当配置为 2 位时， $ADDR\langle 11\rangle$ 使能无效，译出 16 位中有连续 2 位有效，能选择 16 片中连续 2 片。当配置为 4 位时，译出 16 位中有连续 4 位有效，能选择 16 片中连续 4 片。配置为 8 位就能选择 16 片中的上 8 片或下 8 片。配置为 16 位，4 个地址均无效，译出的 16 位全有效，16 片全选。经过了片选的一级译码，列译码还需经过第二级的片内译码。

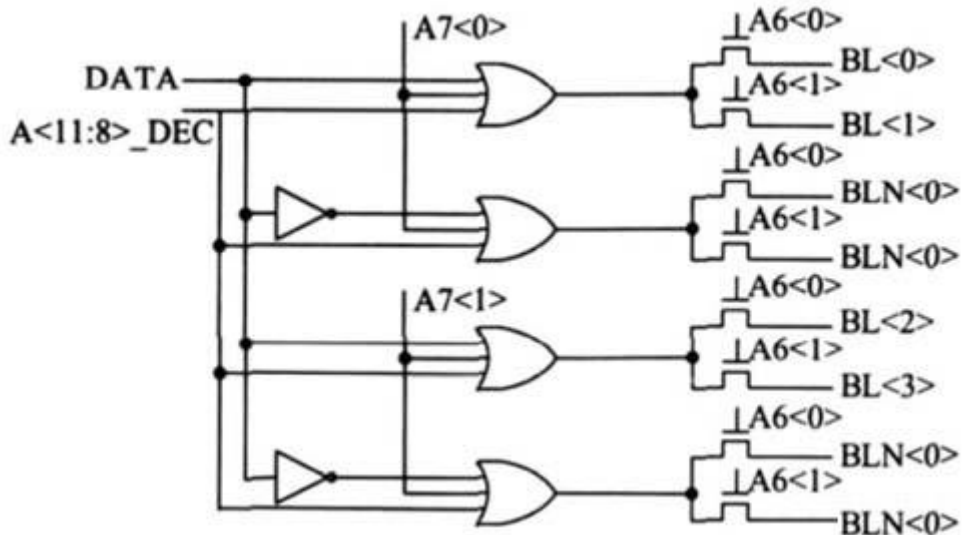


图4 片内译码

$A\langle 11:8\rangle_DEC$ 与 $A7$ 译码均为低有效， $A6$ 译码为高有效。之所以能够用或门译码，是因为没被译码的一对 BL 和 BLN 位线上的数据是不会被写入存储单

元的，如 A7《0》为 1，A《11:8》_DEC 为 1，BL《0》与 BLN《0》均为 1，即使字线打开了，它们也是不会被写入存储阵列的。而被译码选中的一对位线，BL 与 BLN 互补，它们上的数据即可被写入存储单元。

3.2.3 位线充电电路

对位线的充电共有两对充电管和一对上拉管，宽长比在设计上也是有讲究的。上拉管一直开启，为倒比管。栅极接平衡管的 M1 和 M2 时序要求较高，因为它们的宽长比较大，为主要充电管。在 BRAM 总使能信号 ENA 和时钟 CLK 有效时工作，进行预充电。在 CLK 下降沿，M1 和 M2 短暂关闭可执行读操作。M1、M2 和平衡管都在 Pre1_BL 信号控制下工作。

Pre1_BL 需在数据线与位线之间的开关管打开时关闭，不影响数据的读操作。Pre1_BL 信号受到数据线与位线的开关管控制信号 A 的约束，图 4 的结构即可避免 Pre1_BL 与 A 的时序冲突，在 A 有效时，Pre1_BL 无效，且当 A 关闭时，Pre1_BL 延迟开启。

而 M3 和 M4 管则由 Pre2_BL 信号控制，Pre2_BL 由 BRAM 全局信号 ENA、CLK 和 WE 一起控制。由于 BRAM 在进行写操作时，也可镜像地输出写入的数据，即也做了读操作。为了更好地在写入时也读出，且满足频率要求，有必要增加这一充电管。

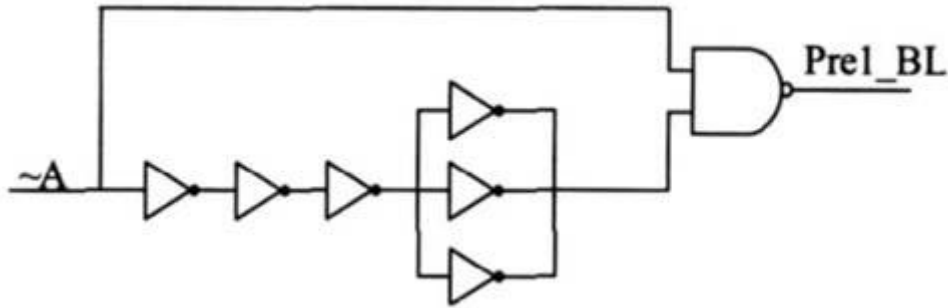


图 5 Pre1_BL 信号产生电路

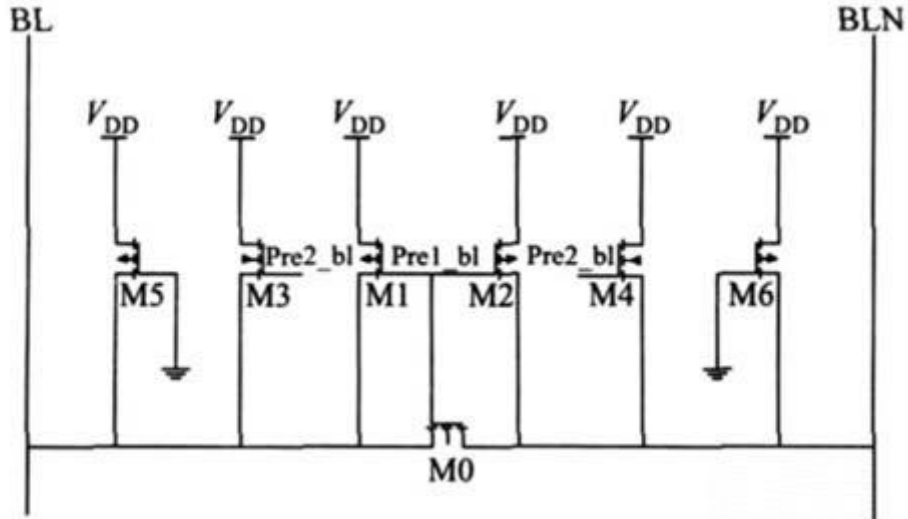


图 6 位线充电电路

4 BRAM 应用

作为随机存取存储器，BRAM 除了实现一般的存储器功能外，还可实现不同数据宽度的存储，且可用作 ROM，以实现组合逻辑函数。当初始化了 BRAM 后，一组地址输入就对应了一组数据的输出，根据数据和地址的对应关系，就能实现一定的函数功能，BRAM 之所以能实现函数逻辑，原因是它拥有足够的存储单元，可以把逻辑函数所有可能的结果预先存入到存储单元中。如实现 4×4 二进制乘法器：

$$Y[3:0] \times X[3:0] = D[7:0]$$

即由地址来查找数据，如同 LUT。在 FPGA 中，还可用 BRAM 来实现 FIFO 中的存储体模块，CLB 实现控制逻辑，设计紧凑，小巧灵活。

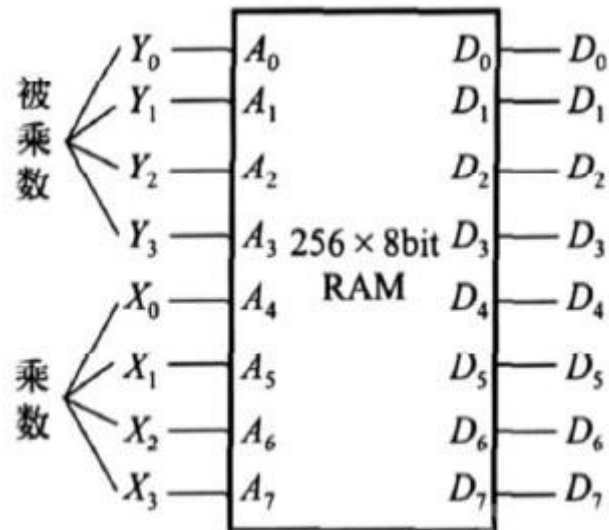


图 7 4 位乘法器

5 结论

如今系统越来越高级，数字电路也高度集成，存储器也越来越多地应用于嵌入式芯片中。本文设计了一种应用于 FPGA 的嵌入式存储器结构，符合一般的双端 SRAM 功能，且具有 FPGA 功能块的可配置选择，灵活性很高。