

本应用笔记介绍了 MAX[®] II 和 MAX V 器件中内部振荡器的例化及其使用。

MAX II 和 MAX V 器件提供了一个独特的内部振荡器作为用户闪存 (UFM) 的一个组成部分。从该应用笔记介绍的设计实例中可以看到，内部振荡器对需要内部时钟的设计提供了绝佳的选择，从而节省了电路板面积以及与外部时钟电路相关的成本。

本应用笔记介绍了如下几部分内容：

- 内部振荡器 (第 1 页)
- 使用 MAX II 和 MAX V 器件的内部振荡器 (第 3 页)
- 设计实现 (第 7 页)
- 源代码 (第 8 页)

内部振荡器

大部分设计在正常工作时都需要时钟。利用内部振荡器，MAX II 和 MAX V 器件不再需要外部时钟电路。例如，内部振荡器能够满足 LCD 控制器、系统管理总线 (SMBus) 控制器以及其他接口协议的时钟要求，还可用以实现脉冲宽度调制器。这将有助于减少元器件数量，节省电路板面积，并降低总的系统成本。

通过使用 Quartus[®] II 软件的 MAX II/MAX V 振荡器宏功能，不需要例化 UFM 就可以例化内部振荡器。

振荡器输出频率 *osc* 是内部振荡器未分频率的四分之一。表 1 列出了 MAX II 和 MAX V 器件中未分频内部振荡器以及振荡器输出的工作频率范围。

表 1. 在 MAX II 和 MAX V 器件中的工作频率范围

参数	工作频率范围 (MHz)	
	MAX II 器件	MAX V 器件
未分频内部振荡器频率	13.33–22.22	15.60–21.20
振荡器输出频率 <i>osc</i>	3.3–5.5	3.9–5.3

在图 1 中内部振荡器是 UFM 的一部分。

图 1. 内部振荡器是 UFM 的一部分（注释 1, 注释 2）

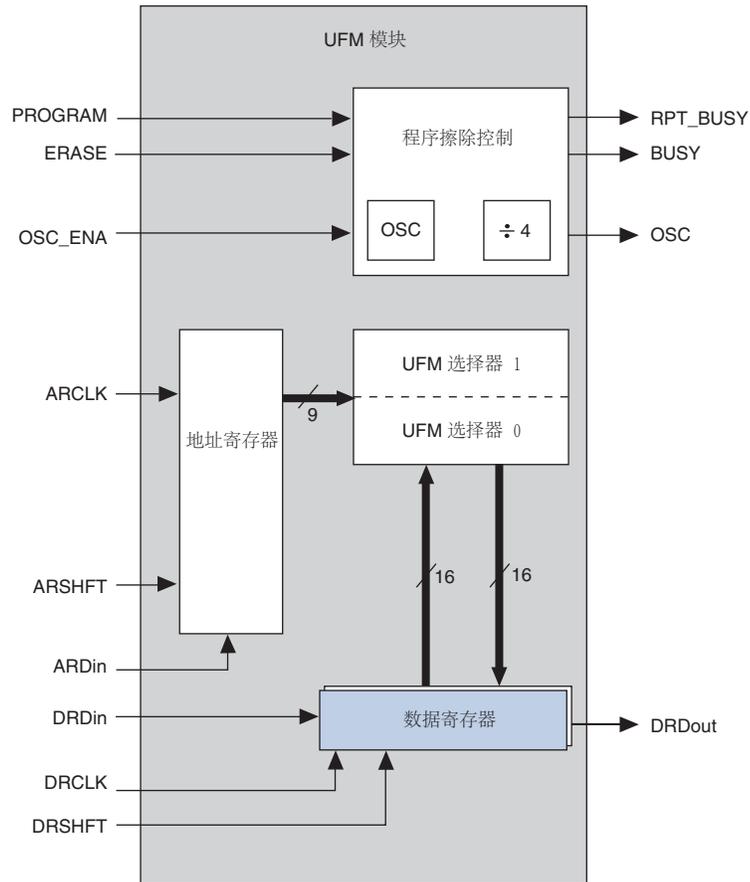


图 1 的注释：

- (1) 内部振荡器是程序擦除控制模块的组成部分，该模块控制 UFM 的编程和擦除。数据寄存器保存即将写入或者刚从 UFM 读取的数据。而地址寄存器则保存已经读取或即将写入的数据地址。
- (2) 当执行 ERASE, PROGRAM 和 READ 操作时，UFM 模块的内部振荡器开始工作。

表 2 列出了 MAX II 和 MAX V 振荡器宏功能中使用的信号。

表 2. MAX II 和 MAX V 振荡器宏功能中引脚说明

信号	说明
OSC_ENA	用于使能内部振荡器信号。
OSC	内部振荡器输出。振荡器没有工作时，该信号为低电平。

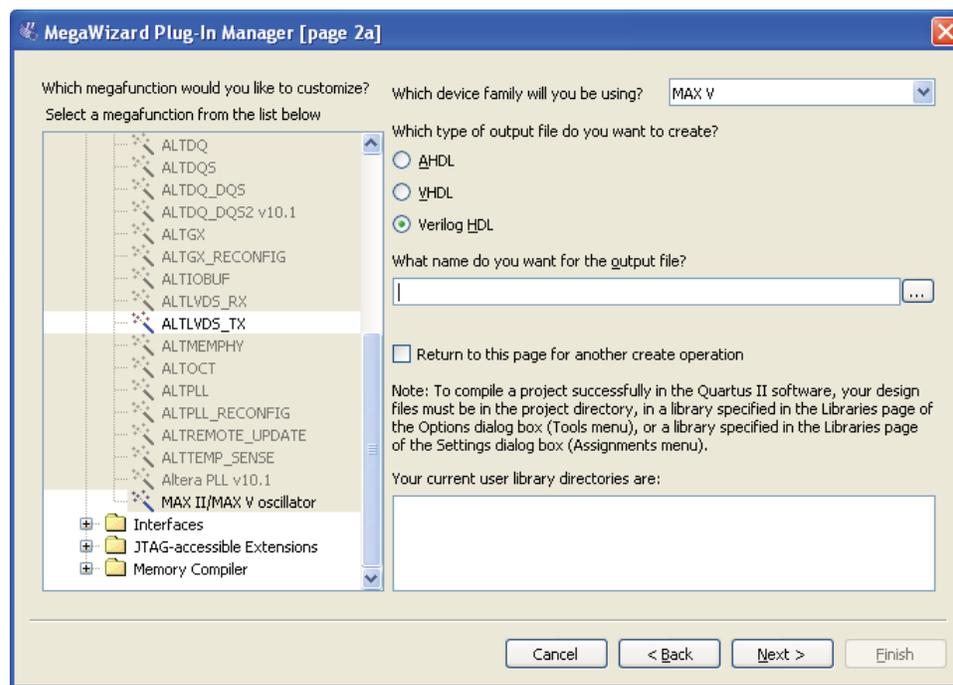
使用 MAX II 和 MAX V 器件的内部振荡器

内部振荡器有一路输入 OSC_ENA 和一路输出 OSC。输入引脚 OSC_ENA 用于激活内部振荡器。激活后，输出上产生的频率为第 1 页表 1 中所示的时钟信号。如果 OSC_ENA 被驱动为低电平，内部振荡器的输出保持高电平不变。

可以使用 MegaWizard™ 插件管理器的 MAX II/MAX V 振荡器宏功能来例化内部振荡器，按下面步骤进行：

1. 打开要例化内部振荡器的工程。
2. 在 **Tools** 菜单中，单击 **MegaWizard Plug-In Manager**。
3. 在 MegaWizard 插件管理器的第一页，选择 **Create a new custom megafunction variation**，然后单击 **Next**。
4. 在 MegaWizard 插件管理器第 2a 页，选择 **MAX V**（或者 **MAX II**）以及文件输出类型（图 2）。

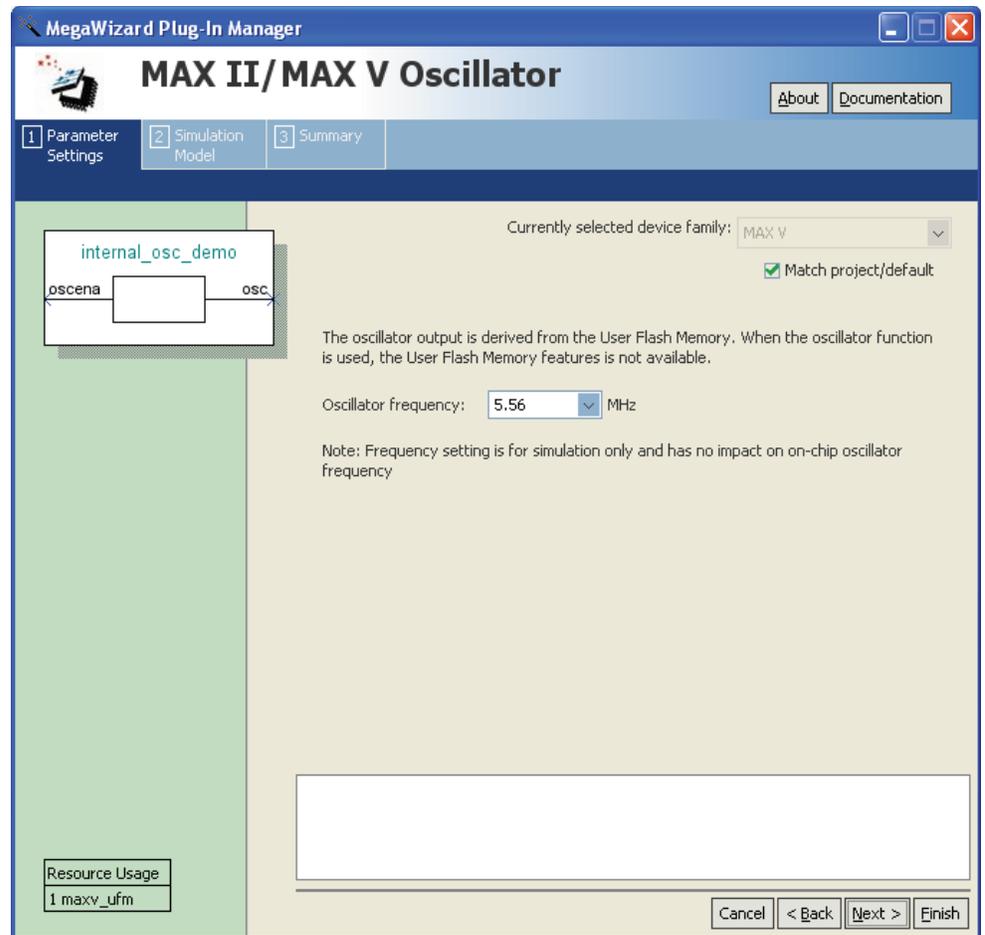
图 2. 在 MegaWizard 插件管理器中选择 ALTUFM_OSC 宏功能



5. 在宏功能列表中，双击 I/O，然后单击 **MAX II/MAX V oscillator**。输入输出文件名，然后单击 **Next**。现在可以选择振荡器输出频率（图 3）。

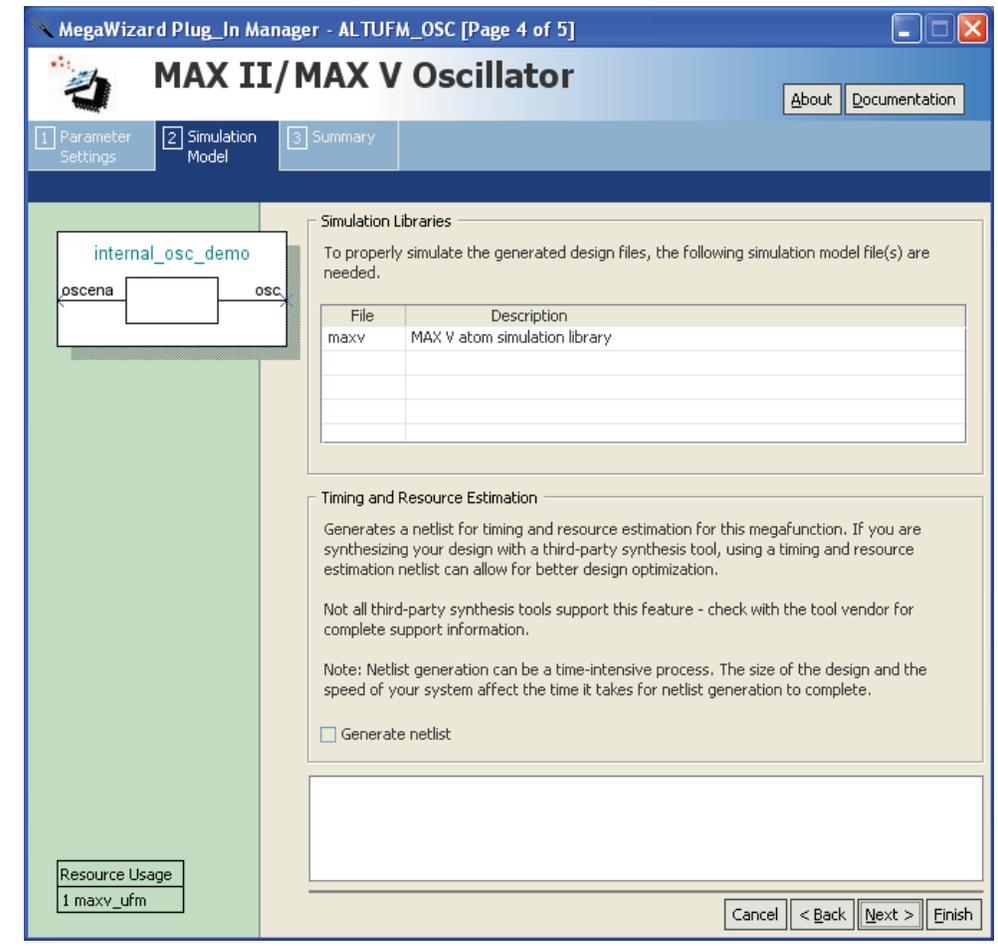
 请注意，频率设置只用于仿真，对片内振荡器频率没有影响。实际输出频率请参看第 1 页表 1，此频率决定于 CPLD。

图 3. OSC 宏功能 MegaWizard 插件管理器第 3 页



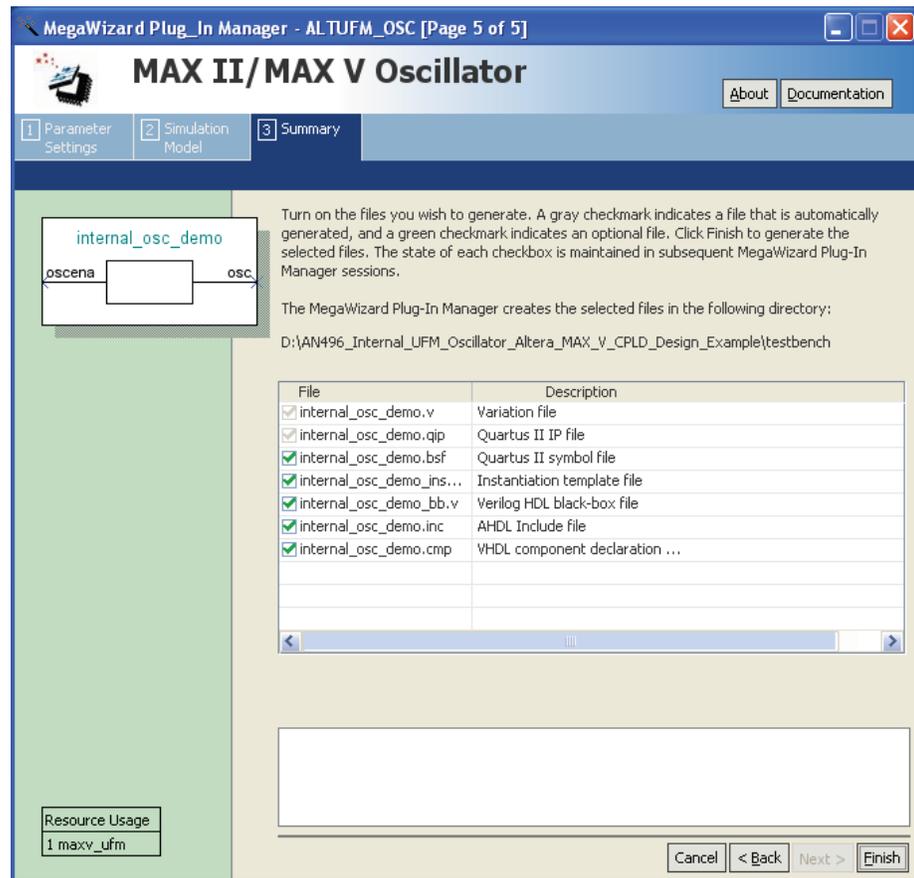
6. 在 **Simulation Libraries** 中，列出了必须包含的模型文件，如图 4 所示。单击 **Next**。

图 4. 仿真库



7. 选择要产生的文件（图 5）。单击 **Finish**。

图 5. OSC 宏功能 MegaWizard 插件管理器总结页面



被选择的文件将被生成于 outputfile 文件夹中（第 3 页图 2）。一旦将例化代码加入到文件后，OSC_ENA 输入必须作为连线型变量，分配逻辑值“1”，以启用振荡器。

设计实现

该设计实例可以采用 MAX II 和 MAX V 器件来实现，这些器件都含有内部振荡器。其具体实现包括将振荡器输出分配给计数器，驱动 MAX II 和 MAX V 器件上的通用 I/O (GPIO) 引脚，以演示内部振荡器的功能。

设计实例 1：使用 MDN-82 演示板 (MAX II 器件)

通过设计实例 1 驱动 LED，产生滚动效应，使用 MDN-82 演示板演示内部振荡器。

表 3 列出了该设计实例的 EPM240G 引脚分配。

表 3. 使用 MDN-82 演示板实现的设计实例 1 的引脚分配

EPM240G 引脚分配			
信号	引脚	信号	引脚
d2	Pin 69	d3	Pin 40
d5	Pin 71	d6	Pin 75
d8	Pin 73	d10	Pin 73
d11	Pin 75	d12	Pin 71
d4_1	Pin 85	d4_2	Pin 69
d7_1	Pin 87	d7_2	Pin 88
d9_1	Pin 89	d9_2	Pin 90
sw9	Pin 82	—	—

在 Quartus II 软件中，将未使用的引脚分配为 **As input tri-stated**。

在 MDN-B2 演示板上演示该设计时，请按照以下步骤进行：

1. 打开演示板电源（使用滑动开关 SW1）。
2. 通过演示板的 JTAG 插头 JP5 和传统编程电缆 (ByteBlaster™ II 或者 USB - Blaster™)，把设计下载到 MAX II CPLD 中。在编程启动前以及启动过程中，保持演示板上 SW4 的按下状态不变。完成后，关断电源，拔下 JTAG 连接器。
3. 观察红色 LED 和双色 LED 上的 LED 滚动顺序。按下演示板上的 SW9，禁止内部振荡器，LED 显示滚动将停止在当前位置上。

设计实例 2: MAX V CPLD 开发套件

在设计实例 2 中，内部振荡器频率除以 2^{21} 之后被送进 2 比特计数器。通过 2 比特计数器的输出来驱动 LED，在 MAX V CPLD 开发套件中演示内部振荡器。

表 4 列出了设计实例 2 的引脚分配。

表 4. 使用 MAX V CPLD 开发套件实现的设计实例 2 的引脚分配

5M570Z 引脚分配			
信号	引脚	信号	引脚
pb0	M9	LED [0]	P4
osc	M4	LED [1]	R1
clk	P2	—	—

在 MAX V 开发套件中演示该设计时，请按照下面几个步骤进行：

1. 将 USB 电缆插入到 USB 连接器使器件上电。
2. 通过内嵌的 USB - Blaster 将设计下载到 MAX II CPLD 中。
3. 观察闪烁的 LED (LED [0] 和 LED [1])。按下演示板上的 pb0，禁止内部振荡器，闪烁的 LED 将停止在当前位置上。

源代码

设计实例 1 和 2 采用了 Verilog 来实现。正如该应用笔记所介绍的，这两个设计实例分别成功地运行在 MDN-B2 和 MAX V CPLD 开发套件上。

 要下载 MAX II 和 MAX V 设计实例，请参考 [AN 469 设计实例 1](#) 和 [AN 469 设计实例 2](#)。

文档修订历史

表 5 列出了该应用笔记的修订历史。

表 5. 文档修订历史

日期	版本	进行的修改
2011 年 1 月	2.0	加入 MAX V 器件。
2007 年 12 月	1.0	初次发布。