

This application note describes instantiating the internal oscillator and using it in the MAX<sup>®</sup> II and MAX V devices.

MAX II and MAX V devices offer a unique internal oscillator feature as part of the user flash memory (UFM). As shown in the design examples described in this application note, internal oscillators make an excellent choice to implement designs that require clocking, thereby saving on-board space and costs associated with external clocking circuitry.

This application note contains the following sections:

- “Internal Oscillators” on page 1
- “Using the Internal Oscillator in MAX II and MAX V Devices” on page 3
- “Implementation” on page 7
- “Source Code” on page 8

## Internal Oscillators

Most designs require a clock for normal operation. With an internal oscillator, MAX II and MAX V devices do not require external clocking circuitry. For example, you can use the internal oscillator to meet the clocking requirement of an LCD controller, system management bus (SMBus) controller, or any other interfacing protocol, or to implement a pulse width modulator. This helps minimize component count, board space, and reduces the total cost of the system.

You can instantiate the internal oscillator without instantiating the UFM by using the MAX II/MAX V oscillator megafunction in the Quartus<sup>®</sup> II software.

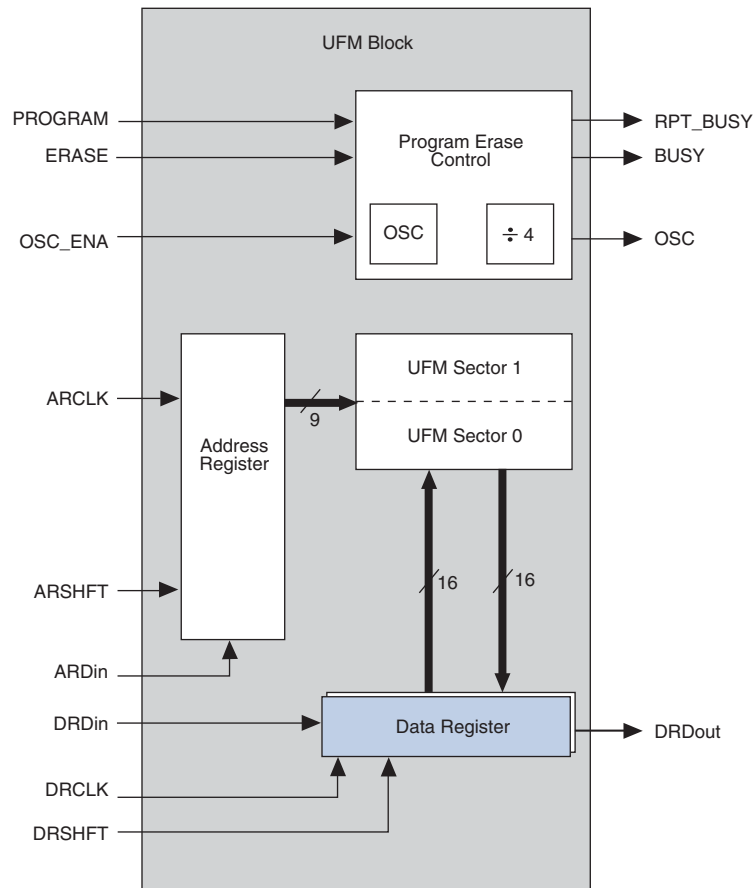
The oscillator’s output frequency, *OSC*, is one-fourth of the undivided frequency of the internal oscillator. Table 1 lists the frequency range of the undivided frequency and oscillator output for MAX II and MAX V devices.

**Table 1. Frequency Range for MAX II and MAX V Devices**

Parameter	Frequency Range (MHz)	
	MAX II Devices	MAX V Devices
Frequency for undivided internal oscillator	13.33–22.22	15.60–21.20
Oscillator’s output frequency, <i>OSC</i>	3.3–5.5	3.9–5.3

Figure 1 shows the internal oscillator as part of the UFM.

**Figure 1. Internal Oscillator as Part of the UFM (Note 1), (2)**



**Notes to Figure 1:**

- (1) The internal oscillator is part of the Program Erase Control block, which controls the programming and erasing of the UFM. The Data Register holds the data to be sent or retrieved from the UFM. The Address Register holds the address from which data is retrieved or the address to which the data is written.
- (2) The internal oscillator for the UFM block is enabled when the ERASE, PROGRAM, and READ operation is executed.

Table 2 lists the signals used in the oscillator megafunction for MAX II and MAX V devices.

**Table 2. Pin Description for the Oscillator Megafunction in MAX II and MAX V Devices**

Signal	Description
OSC_ENA	Use to enable the internal oscillator.
OSC	Output of the internal oscillator. This signal is low when the oscillator is not enabled.

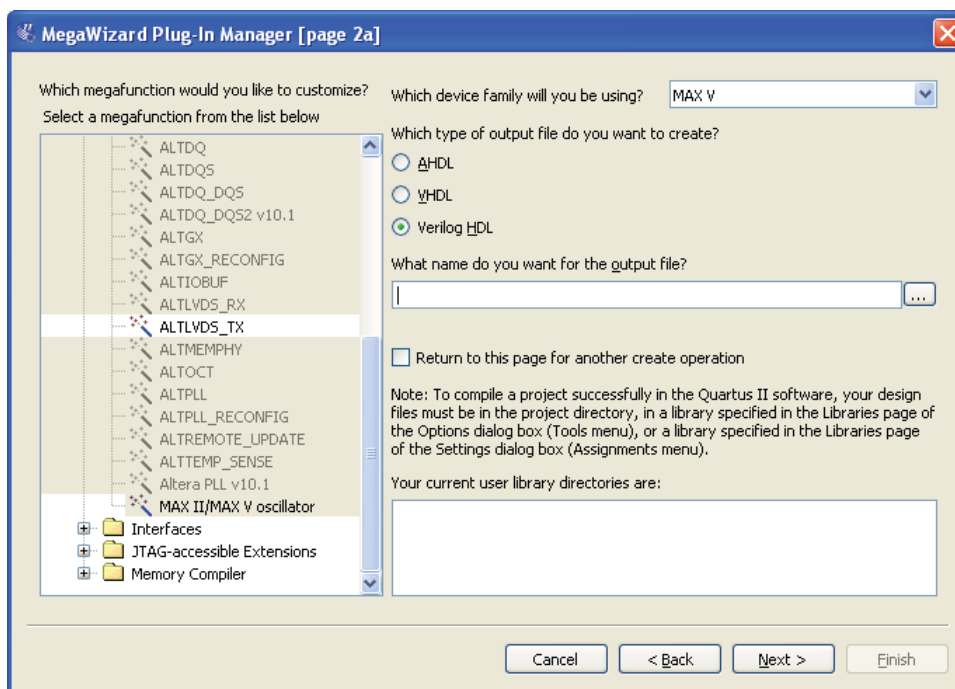
## Using the Internal Oscillator in MAX II and MAX V Devices

The internal oscillator has a single input, `OSC_ENA`, and a single output, `OSC`. To activate the internal oscillator, use `OSC_ENA`. When activated, a clock with the frequency as listed in [Table 1 on page 1](#) is made available at the output. If `OSC_ENA` is driven low, the output of the internal oscillator is a constant high.


To instantiate the internal oscillator, use the MAX II/MAX V oscillator megafunction in the MegaWizard™ Plug-In Manager and follow these steps:

1. Open the project in which the internal oscillator is to be instantiated.
2. On the Tools menu, click **MegaWizard Plug-In Manager**.
3. On page 1 of the MegaWizard Plug-In Manager, select **Create a new custom megafunction variation** and click **Next**.
4. On page 2a of the MegaWizard Plug-In Manager, select **MAX V** (or **MAX II**) and the file output type (refer to [Figure 2](#)).

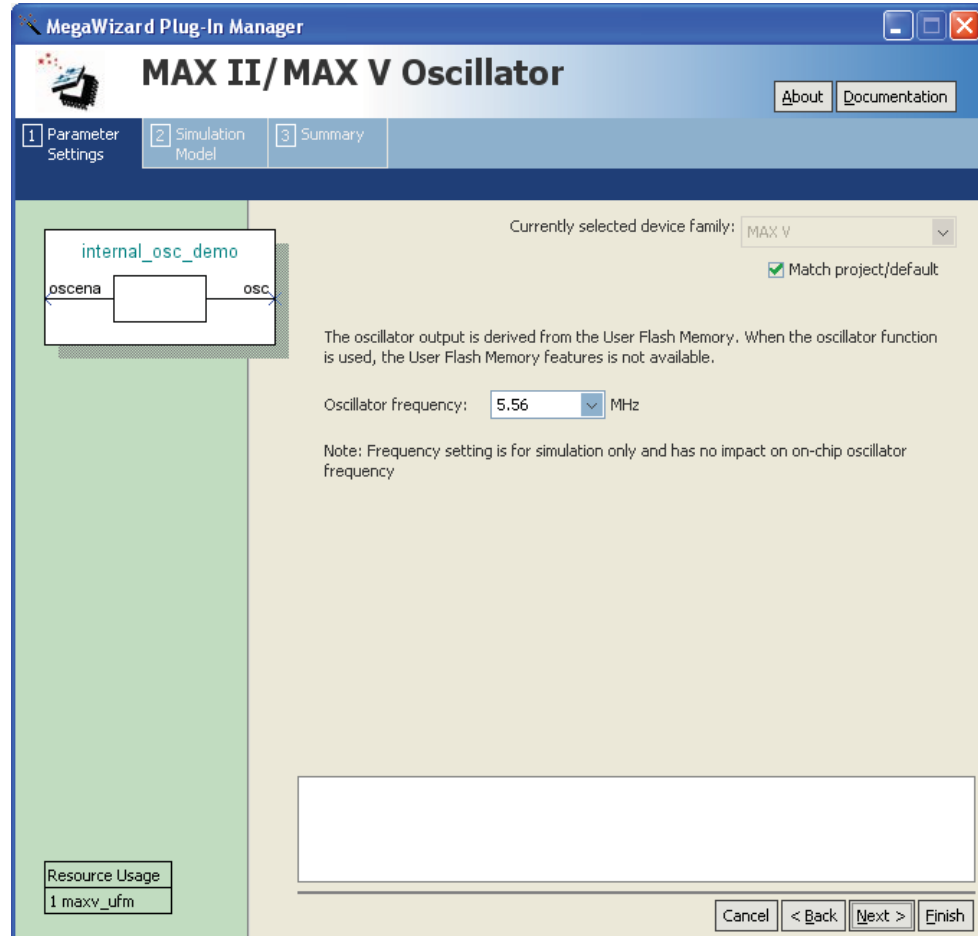
**Figure 2. Selecting the ALTUFM\_OSC Megafunction in the MegaWizard Plug-In Manager**



5. In the Megafunctions list, double-click **I/O** and then click **MAX II/MAX V oscillator**. Type the output file name and click **Next**. You can now select the oscillator output frequency (refer to [Figure 3](#)).

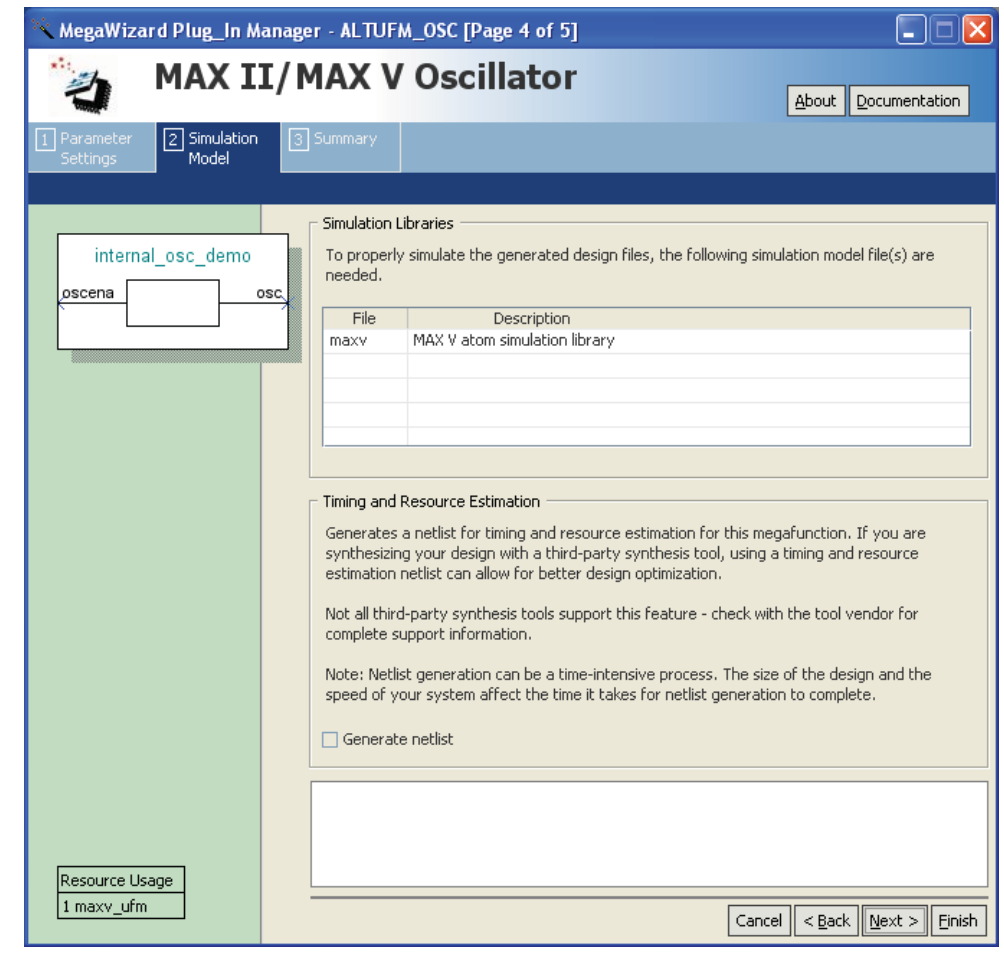
 This frequency setting is only for simulation and has no impact on the on-chip oscillator frequency. This is determined by the CPLD and the output frequency listed in [Table 1 on page 1](#).

**Figure 3. Page 3 of the OSC Megafunction in the MegaWizard Plug-In Manager**



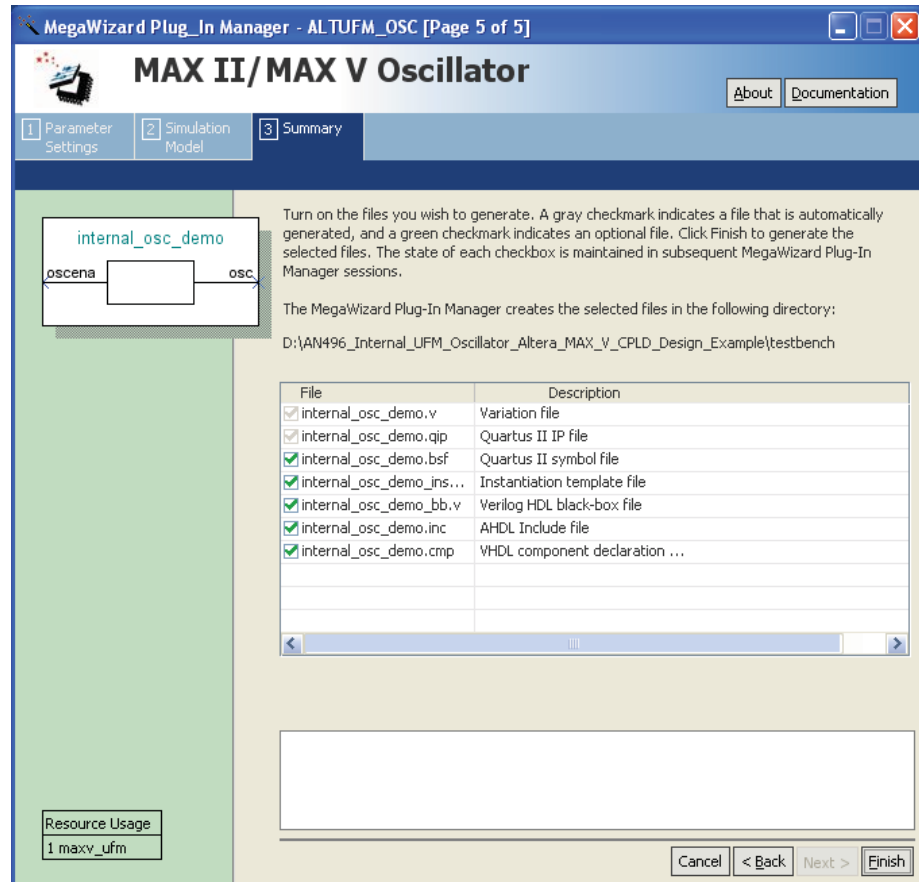
6. In **Simulation Libraries**, the model files that must be included are listed, as shown in [Figure 4](#). Click **Next**.

**Figure 4. Simulation Libraries**



7. Select the files to be created (refer to [Figure 5](#)). Click **Finish**.

**Figure 5. Summary Page of the OSC Megafunction MegaWizard Plug-In Manager**



The selected files are created and can be accessed from the output file folder (refer to [Figure 2 on page 3](#)). After the instantiation code is added to the file, the `OSC_ENA` input must be made as a wire and assigned as a logic value of "1" to enable the oscillator.

## Implementation

You can implement this design example with MAX II and MAX V devices, all of which have the internal oscillator feature. Implementation involves demonstration of the internal oscillator function by assigning the oscillator output to a counter and driving the general purpose I/O (GPIO) pins on MAX II and MAX V devices.

### Design Example 1: Targeting a MDN-82 Demo Board (MAX II Devices)

Design Example 1 is made to drive the LEDs to create a scrolling effect, thereby demonstrating the internal oscillator using the MDN-82 demo board.

Table 3 lists the EPM240G pin assignments for Design Example 1.

**Table 3. Pin Assignments for Design Example 1 Using the MDN-82 Demo Board**

EPM240G Pin Assignments			
Signal	Pin	Signal	Pin
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	—	—

Assign the unused pins **As input tri-stated** in the Quartus II software.

To demonstrate this design on the MDN-B2 demo board, follow these steps:

1. Turn on the power to the demo board (using slide switch SW1).
2. Download the design onto the MAX II CPLD through the JTAG header JP5 on the demo board and a conventional programming cable (ByteBlaster™ II or USB-Blaster™). Keep SW4 on the demo board pressed before and during the start of the programming process. After it completes, turn off the power and remove the JTAG connector.
3. Observe the scrolling LED sequence on the red LEDs and the bi-color LEDs. Pressing SW9 on the demo board disables the internal oscillator and the scrolling LEDs will freeze at their current positions.

## Design Example 2: Targeting a MAX V CPLD Development Kit

In Design Example 2, the oscillator output frequency is divided by  $2^{21}$  before clocking a 2-bit counter. The output of this 2-bit counter is used to drive the LEDs, thereby demonstrating the internal oscillator on the MAX V CPLD development kit.

Table 4 lists the 5M570Z pin assignments for Design Example 2.

**Table 4. Pin Assignments for Design Example 2 Using the MAX V CPLD Development Kit**

5M570Z Pin Assignments			
Signal	Pin	Signal	Pin
pb0	M9	LED [0]	P4
osc	M4	LED [1]	R1
clk	P2	—	—

To demonstrate this design on the MAX V development kit, follow these steps:

1. Plug in the USB cable into the USB Connector to power up the device.
2. Download the design onto the MAX V CPLD through the embedded USB-Blaster.
3. Observe the blinking LEDs (LED [0] and LED [1]). Pressing pb0 on the demo board disables the internal oscillator and the blinking LEDs will freeze at their current state.

## Source Code

Design Examples 1 and 2 were implemented in Verilog. Successful operations have been demonstrated using the MDN-B2 demo board and MAX V CPLD Development Kit respectively, as described in this application note.



To download the MAX II and MAX V design examples, refer to [AN 469 Design Example 1](#) and [AN 469 Design Example 2](#).

## Document Revision History

Table 5 lists the revision history for this application note.

**Table 5. Document Revision History**

Date	Version	Changes
January 2011	2.0	Updated to include MAX V devices.
December 2007	1.0	Initial release.