

## 6601/6602

### Introduction

---

This document contains information and step-by-step instructions for verifying the operation of the crystal oscillator in the National Instruments 6601/6602 data acquisition devices.

### What Is Calibration?

*Calibration* consists of verifying the measurement accuracy of a device and correcting for any measurement error. For 6601/6602 devices, calibration is simply verifying the measurement accuracy of the crystal oscillator on the device, because the crystal oscillator is not user-adjustable. *Verification* is measuring the performance of a device and comparing the results to the factory specifications of the device. If the procedure finds the device is not operating within its specifications, contact National Instruments for a replacement device.

### Why Should You Verify?

The accuracy of timing components can drift with time and temperature, which could cause them to be less accurate than specified. Verification determines whether your device still meets National Instruments standards.

### How Often Should You Verify?

The measurement requirements of your application determine how often you should verify the performance of your device to ensure its accuracy. National Instruments recommends that you verify your device at least once every year. You can shorten this interval to six months or 90 days, based on the demands of your application.

# Equipment and Other Test Requirements

---

This section describes the equipment, software, documentation, and test conditions required for verifying the operation of 6601/6602 devices.

## Test Equipment

Verifying the 6601/6602 devices requires the following equipment:

- A highly stable 10 MHz, 5 MHz, or 1 MHz clock source that is accurate to within 1 ppm. National Instruments recommends using the PXI-6608 to provide the stable clock. The PXI-6608 provides a 10 MHz clock that is accurate to 0.075 ppm over 1 year.
- An SH68-68-D1 cable or a standard 68-pin cable connector
- A 68-pin breakout box (if you are using a clock source other than the PXI-6608)

## Software and Documentation

You will need the following software and documentation to verify the 6601/6602. You can find these on the National Instruments Web site at [ni.com](http://ni.com):

- The latest version of the National Instruments NI-DAQ driver
- *NI-DAQ Function Reference Online Help* file
- *NI-DAQ User Manual for PC Compatibles*

### Software

The 6601/6602 calibration procedure requires using the latest version of the National Instruments NI-DAQ driver software. NI-DAQ supports a number of application development environments (ADEs) and programming languages, including LabVIEW, LabWindows/CVI, Microsoft Visual C++, Microsoft Visual Basic, and Borland C++. When you install the driver, you only need to install support for the ADE or programming language you will use.

### Documentation

The *NI-DAQ Function Reference Online Help* file and *NI-DAQ User Manual for PC Compatibles* contain detailed information on using the NI-DAQ driver. The online help file includes detailed information on the driver functions. You can access the online help file by clicking **Start»Programs»National Instruments»NI-DAQ»NI-DAQ Help**. The *NI-DAQ User Manual for PC Compatibles* provides instructions on installing NI-DAQ. This manual also includes detailed information on creating applications that use the NI-DAQ driver. These are your primary

references for writing your calibration utility. You can also refer to the documentation for the programming language you are using. For further information on the products you are calibrating, refer to the *660X User Manual*.

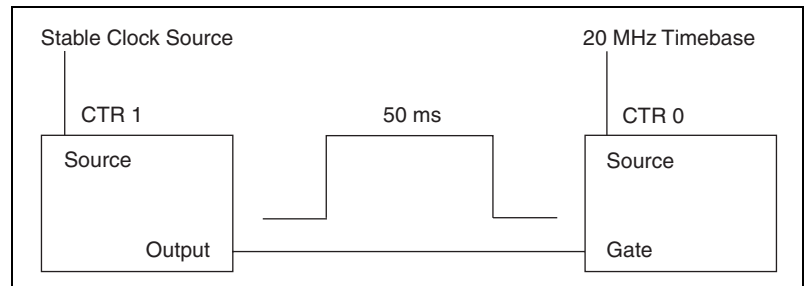
## Test Conditions

Follow these guidelines to optimize connections and test conditions during the verification procedure:

- Keep connections to the device as short as possible. Long cables and wires act as antennae by picking up extra noise that can affect measurements.
- Maintain a temperature of approximately 25 °C.
- Keep relative humidity below 80%.
- Use shielded copper wire for all cable connections to the device. Use twisted-pair wire to eliminate noise and thermal offsets.
- Allow a warm-up time of at least 15 minutes to ensure the measurement circuitry is at a stable operating temperature.

## Verifying the Operation of 6601/6602 Devices

This section contains instructions for verifying the operation of the crystal oscillator on 6601/6602 devices using a stable clock source. Verifying the performance of the oscillator on a 6601/6602 device requires two counters on the 6601/6602 device—counters 0 and 1. Counter 1 uses your stable clock as the source to generate a 50 ms single pulse, which is applied to the GATE of counter 0. The 20 MHz timebase on the 6601/6602 device, which is derived from the crystal oscillator, is the SOURCE of counter 0. You will count the number of clock edges of the 20 MHz timebase that occur during the 50 ms single pulse and calculate the frequency of the oscillator to determine if the device is within its specifications. Figure 1 shows a simple diagram of the process.



**Figure 1.** 6601/6602 Verification Process

The frequency measurement will have an uncertainty that is equal to the sum of the stable clock error and the quantization error caused by the counter. The clock error should be 1 ppm or less, as specified in the [Test Equipment](#) section of this document. The quantization error, which is the inherent uncertainty in digitizing an analog value, will always be up to 50 ns. Using a measurement duration of 50 ms gives a quantization error as follows:

$$\frac{50 \text{ ns}}{50 \text{ ms}} = 1 \text{ ppm}$$

The sum of the stable clock error, 1 ppm, and the quantization error, 1 ppm, gives a total measurement uncertainty of 2 ppm. You can reduce measurement uncertainty by using a more stable clock, by increasing the measurement duration, or both. The calculations presented at the end of this document allow for a 2 ppm measurement uncertainty.

Both NI-DAQ programming and LabVIEW programming instructions are presented in the following sections. Choose the appropriate instructions for your programming environment.

## Setting Up the Stable Clock

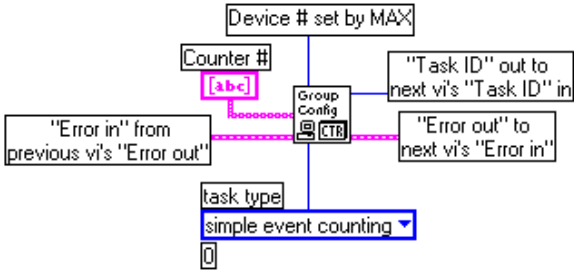
This section describes how to set up your clock. National Instruments recommends using the PXI-6608 to provide the stable clock. You can use an alternative stable clock if it meets the specifications described in the [Test Equipment](#) section of this document. Instructions for both clock alternatives are included in the following two sections, [Setting Up a PXI-6608 Clock Source](#) and [Setting Up an Alternative Clock Source](#).

### Setting Up a PXI-6608 Clock Source

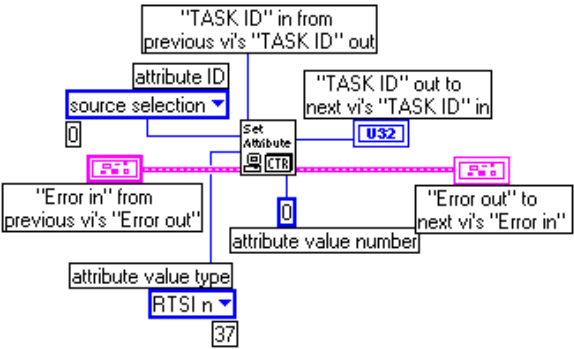
To set up a PXI-6608 as your stable clock source, complete the following steps:

1. Power the 6608 device.
2. Wait 15 minutes to allow the measurement circuitry in the device to warm up.

3. Route the 10 MHz stable clock of the PXI-6608 onto PFI\_35.
  - a. Configure counter 1 on the PXI-6608 to perform simple event counting.

NI-DAQ Function Call	LabVIEW Block Diagram
<p>Call GPCTR_Set_Application with the following parameters:</p> <p><b>deviceNumber:</b> The value set by Measurement &amp; Automation Explorer (MAX)</p> <p><b>gpctrNum:</b> ND_COUNTER_1</p> <p><b>application:</b> ND_SIMPLE_EVENT_CNT</p>	 <p>The LabVIEW block diagram shows the configuration of a counter. It starts with a 'Device # set by MAX' block. This connects to a 'Counter #' block with a value of '1'. The 'Counter #' block is connected to a 'Group Config' block (CTB). The 'Group Config' block is also connected to a 'task type' dropdown menu set to 'simple event counting'. The 'Group Config' block has two error handling outputs: 'Error in' from previous vi's 'Error out' and 'Error out' to next vi's 'Error in'. The 'Group Config' block is also connected to a 'Task ID' output block labeled 'Task ID' out to next vi's 'Task ID' in'.</p>

- b. Configure the **source** of counter 1 as RTSI line 0.

NI-DAQ Function Call	LabVIEW Block Diagram
<p>Call GPCTR_Change_Parameter with the following parameters:</p> <p><b>deviceNumber:</b> The value set by MAX</p> <p><b>gpctrNum:</b> ND_COUNTER_1</p> <p><b>paramID:</b> ND_SOURCE</p> <p><b>paramValue:</b> ND_RTSI_0</p>	 <p>The LabVIEW block diagram shows the configuration of the counter source. It starts with a 'TASK ID' input block labeled 'TASK ID' in from previous vi's 'TASK ID' out'. This connects to an 'attribute ID' block with a value of '0'. The 'attribute ID' block is connected to a 'source selection' dropdown menu set to '0'. The 'source selection' block is connected to a 'Set Attribute' block (CTB). The 'Set Attribute' block is also connected to a 'U32' block with a value of '0'. The 'Set Attribute' block has two error handling outputs: 'Error in' from previous vi's 'Error out' and 'Error out' to next vi's 'Error in'. The 'Set Attribute' block is also connected to an 'attribute value number' block with a value of '0'. The 'Set Attribute' block is also connected to an 'attribute value type' dropdown menu set to 'RTSI n'. The 'attribute value type' block is connected to a '37' block.</p>

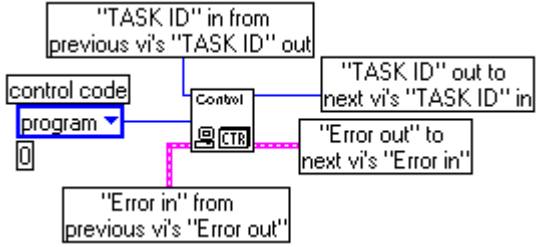
- c. Route the stable 10 MHz clock on the PXI-6608 onto RTSI line 0.

NI-DAQ Function Call	LabVIEW Block Diagram
<p>Call <code>Select_Signal</code> with the following parameters:</p> <p><b>deviceNumber:</b> The value set by MAX</p> <p><b>signal:</b> ND_RTSI_0</p> <p><b>source:</b> ND_STABLE_10_MHZ</p> <p><b>sourceSpec:</b> ND_DONT_CARE</p>	

- d. Configure the **source** of counter 1 to be available on PFI line 35.

NI-DAQ Function Call	LabVIEW Block Diagram
<p>Call <code>Select_Signal</code> with the following parameters:</p> <p><b>deviceNumber:</b> The value set by MAX</p> <p><b>signal:</b> ND_PFI_35</p> <p><b>source:</b> ND_GPCTR1_SOURCE</p> <p><b>sourceSpec:</b> ND_DONT_CARE</p>	

- e. Program counter 1 to arm it.

NI-DAQ Function Call	LabVIEW Block Diagram
<p>Call GPCTR_Control with the following parameters:</p> <p><b>deviceNumber:</b> The value set by MAX</p> <p><b>gpctrNum:</b> ND_COUNTER_1</p> <p><b>action:</b> ND_PROGRAM</p>	 <p>“TASK ID” out connects to “TASK ID” in of step 14 in the <i>Measuring the Frequency of 6601/6602 Device</i> section.</p>

## Setting Up an Alternative Clock Source

To set up a clock source other than a PXI-6608, complete the following steps:

1. Ensure that your clock source meets the requirements listed in the *Test Equipment* section.
2. Power the clock device.



**Note** After you complete the clock setup procedure, your clock is running and generating pulses.

## Connecting the Clock Source to the 6601/6602 Device

This section describes how to connect your clock source to the 6601/6602 device, whether you are using a PXI-6608 or another clock source.

### Connecting a PXI-6608 to Your 6601/6602 Device

Connect the PXI-6608 clock source to your 6601/6602 device using an SH68-68-D1 cable, which connects to the I/O connector of each device.

# Connecting an Alternative Clock Source to Your 6601/6602 Device

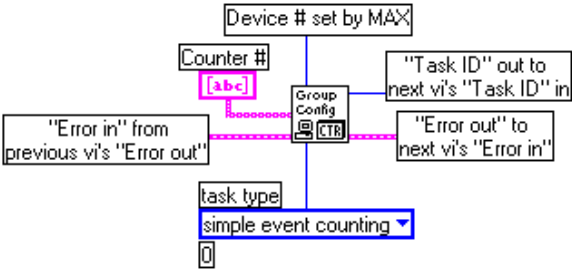
To connect an alternative clock source to your 6601/6602 device, complete the following steps:

1. Connect your 6601/6602 device to a 68-pin breakout box using the SH68-68-D1 cable.
2. Connect your clock source to pins 7 and 41 of the 68-pin breakout box using twisted-pair wires.
  - Connect your clock to pin 7. Pin 7 corresponds to PFI\_35, which is the SOURCE of counter 1 on all 6601/6602 devices.
  - Connect the ground of your clock to pin 41, which is digital ground.

## Measuring the Frequency of 6601/6602 Device

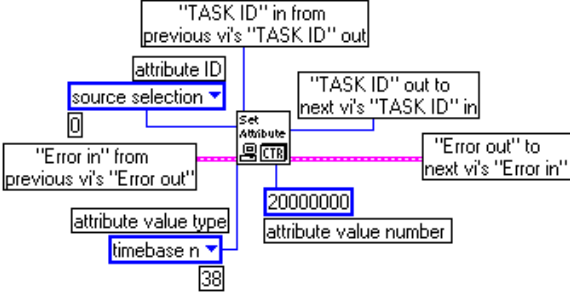
To measure the frequency of your 6601/6602 device for verification, complete the following steps:

1. Configure counter 0 to perform simple event counting.

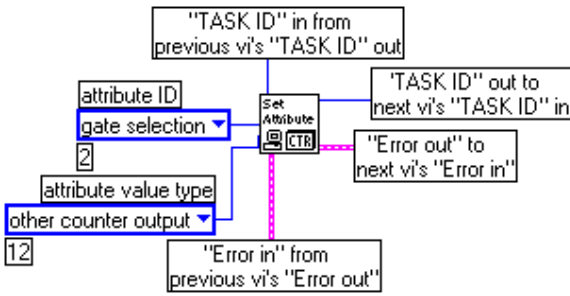
NI-DAQ Function Call	LabVIEW Block Diagram
<p>Call GPCTR_Set_Application with the following parameters:</p> <p><b>deviceNumber:</b> The value set by MAX</p> <p><b>gpctrNum:</b> ND_COUNTER_0</p> <p><b>application:</b> ND_SIMPLE_EVENT_CNT</p>	 <p>The LabVIEW block diagram shows the configuration for GPCTR_Set_Application. It includes a 'Device # set by MAX' control, a 'Counter #' control with a value of '1', and a 'task type' dropdown menu set to 'simple event counting'. The 'Group Config' block is connected to the 'Counter #' control and the 'task type' dropdown. The 'Error in' and 'Error out' terminals are connected to the 'Group Config' block, with the 'Error in' terminal receiving input from a previous vi's 'Error out' and the 'Error out' terminal sending output to the next vi's 'Error in'. The 'Task ID' terminal is also connected to the 'Group Config' block, with the 'Task ID' output going to the next vi's 'Task ID' input.</p>



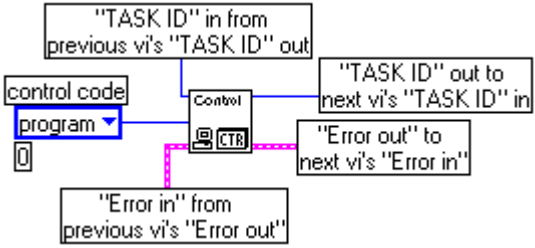
- Set the **source** of counter 0 to be the 20 MHz timebase.

NI-DAQ Function Call	LabVIEW Block Diagram
<p>Call GPCTR_Change_Parameter with the following parameters:</p> <p><b>deviceNumber:</b> The value set by MAX</p> <p><b>gpctrNum:</b> ND_COUNTER_0</p> <p><b>paramID:</b> ND_SOURCE</p> <p><b>paramValue:</b> ND_INTERNAL_20_MHZ</p>	

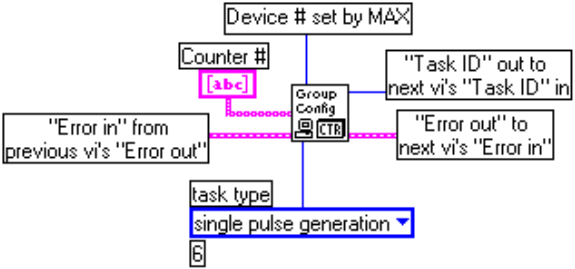
- Configure the **gate** of counter 0 to be the **output** of counter 1.

NI-DAQ Function Call	LabVIEW Block Diagram
<p>Call GPCTR_Change_Parameter with the following parameters:</p> <p><b>deviceNumber:</b> The value set by MAX</p> <p><b>gpctrNum:</b> ND_COUNTER_0</p> <p><b>paramID:</b> ND_GATE</p> <p><b>paramValue:</b> ND_OTHER_GPCTR_OUTPUT</p>	

4. Arm counter 0 to perform simple event counting.

NI-DAQ Function Call	LabVIEW Block Diagram
<p>Call GPCTR_Control with the following parameters:</p> <p><b>deviceNumber:</b> The value set by MAX</p> <p><b>gpctrNum:</b> ND_COUNTER_0</p> <p><b>action:</b> ND_PROGRAM</p>	 <p>The diagram shows a control loop for counter 0. It starts with a 'control code' block set to 'program' and a numeric control value of 0. These are connected to a 'Control' block with a counter icon. The counter's output is connected to a 'TASK ID' out terminal, which then connects to the 'TASK ID' in terminal of the next step. The counter also has an 'Error out' terminal connected to the 'Error in' terminal of the next step.</p> <p>“TASK ID” out connects to “TASK ID” in of step 11.</p>

5. Configure counter 1 to perform single-pulse generation.

NI-DAQ Function Call	LabVIEW Block Diagram
<p>Call GPCTR_Set_Application with the following parameters:</p> <p><b>deviceNumber:</b> The value set by MAX</p> <p><b>gpctrNum:</b> ND_COUNTER_1</p> <p><b>application:</b> ND_SINGLE_PULSE_GNR</p>	 <p>The diagram shows the configuration for counter 1. It includes a 'Device # set by MAX' block, a 'Counter #' block set to 1, and a 'task type' block set to 'single pulse generation'. These are connected to a 'Group Config' block with a counter icon. The counter's output is connected to a 'Task ID' out terminal, which then connects to the 'Task ID' in terminal of the next step. The counter also has an 'Error out' terminal connected to the 'Error in' terminal of the next step.</p>

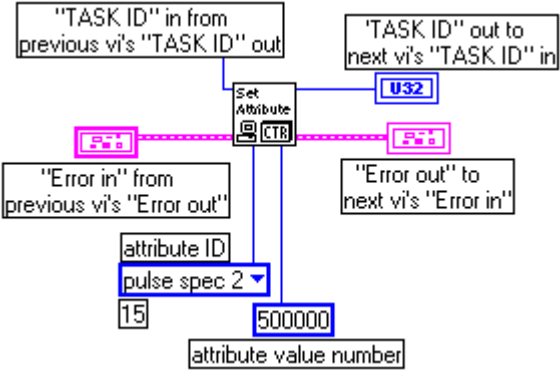
6. Configure **source** of counter 1 to be PFI line 35.

NI-DAQ Function Call	LabVIEW Block Diagram
<p>Call GPCTR_Change_Parameter with the following parameters:</p> <p><b>deviceNumber:</b> The value set by MAX</p> <p><b>gpctrNum:</b> ND_COUNTER_1</p> <p><b>paramID:</b> ND_SOURCE</p> <p><b>paramValue:</b> ND_PFI_35</p>	

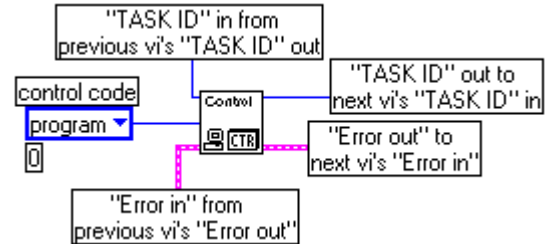
7. Set the pulse delay for the single-pulse generation.

NI-DAQ Function Call	LabVIEW Block Diagram
<p>Call GPCTR_Change_Parameter with the following parameters:</p> <p><b>deviceNumber:</b> The value set by MAX</p> <p><b>gpctrNum:</b> ND_COUNTER_1</p> <p><b>paramID:</b> ND_COUNT_1</p> <p><b>paramValue:</b> 2</p>	

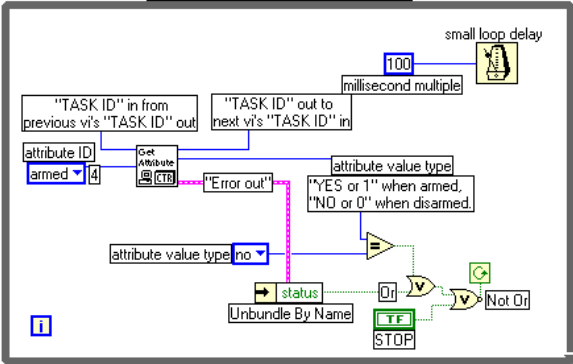
8. Set the pulse width for the single-pulse generation. The following example assumes a 10 MHz source (period = 100 ns) for counter 1. Therefore, the pulse width is  $100 \text{ ns} \times 500000 = 50 \text{ ms}$ . If you use a frequency other than 10 MHz, adjust the number accordingly. For a frequency  $F$ ,  $1/F \times \text{ND\_COUNT\_2}$  should equal 50 ms.

NI-DAQ Function Call	LabVIEW Block Diagram
<p>Call <code>GPCTR_Change_Parameter</code> with the following parameters:</p> <p><b>deviceNumber:</b> The value set by MAX</p> <p><b>gpctrNum:</b> ND_COUNTER_1</p> <p><b>paramID:</b> ND_COUNT_2</p> <p><b>paramValue:</b> 500000</p>	 <p>The LabVIEW block diagram shows the <code>Set Attribute</code> block with the <code>CTR</code> icon. The <code>attribute ID</code> input is set to <code>pulse spec 2</code> with a value of <code>15</code>. The <code>attribute value number</code> input is set to <code>500000</code>. The <code>U32</code> input is set to <code>0x32</code>. The block is connected to the <code>"TASK ID" in from previous vi's "TASK ID" out</code> and <code>"Error in" from previous vi's "Error out"</code> inputs, and outputs to <code>"TASK ID" out to next vi's "TASK ID" in</code> and <code>"Error out" to next vi's "Error in"</code>.</p>

9. Program counter 1 to perform single-pulse generation.

NI-DAQ Function Call	LabVIEW Block Diagram
<p>Call <code>GPCTR_Control</code> with the following parameters:</p> <p><b>deviceNumber:</b> The value set by MAX</p> <p><b>gpctrNum:</b> ND_COUNTER_1</p> <p><b>action:</b> ND_PROGRAM</p>	 <p>The LabVIEW block diagram shows the <code>Control</code> block with the <code>CTR</code> icon. The <code>control code</code> input is set to <code>program</code> with a value of <code>0</code>. The block is connected to the <code>"TASK ID" in from previous vi's "TASK ID" out</code> and <code>"Error in" from previous vi's "Error out"</code> inputs, and outputs to <code>"TASK ID" out to next vi's "TASK ID" in</code> and <code>"Error out" to next vi's "Error in"</code>.</p>

10. Loop until counter 1 has completed pulse generation.

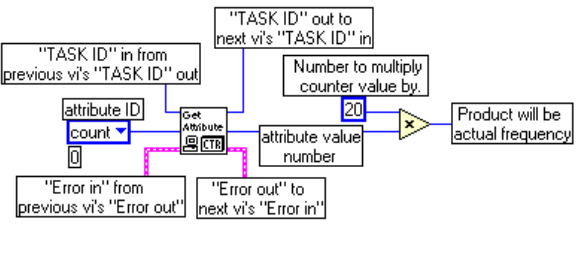
NI-DAQ Function Call	LabVIEW Block Diagram
<p>Call GPCTR_WATCH with the following parameters:</p> <p><b>deviceNumber:</b> The value set by MAX</p> <p><b>gpctrNum:</b> ND_COUNTER_1</p> <p><b>entityID:</b> ND_ARMED</p> <p><b>entityValue:</b> A user-declared variable</p> <p>See example code in the next paragraph.</p>	<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> <p>"WHILE LOOP" will continue until counter is disarmed, an error is returned or the STOP button is pressed.</p> </div>  <p>The LabVIEW block diagram illustrates a while loop that continues until a counter is disarmed, an error is returned, or the STOP button is pressed. The loop contains a 'TASK ID' in terminal that connects to a 'TASK ID' out terminal. The diagram includes a 'Get Attribute' block for 'armed', a 'millisecond multiple' block with a value of 100, and a 'small loop delay' block. The loop continues until the counter is disarmed, an error is returned, or the STOP button is pressed.</p> <p>"TASK ID" out connects to "TASK ID" in of step 13.</p>

**Example Code for GPCTR Watch**

```

i16 iStatus = 0;
u32 ulArmed = ND_YES;
do
{
iStatus = GPCTR_Watch (deviceNumber, ND_COUNTER_1, ND_Armed, &ulArmed);}
while ((ulArmed==ND_YES) && (iStatus==0));
    
```

11. Calculate the frequency of your device.
  - a. Retrieve the count from counter 0. The count represents the number of cycles of the 20 MHz timebase in 50 ms.
  - b. Multiply the count returned above by 20 to calculate the number of pulses that should occur in one second, ( $50 \text{ ms} \times 20 = 1 \text{ s}$ ), which is the *measured frequency* of the 20 MHz timebase.

NI-DAQ Function Call	LabVIEW Block Diagram
<p>Call GPCTR_Watch with the following parameters:</p> <p><b>deviceNumber:</b> The value set by MAX</p> <p><b>gpctrNum:</b> ND_COUNTER_0</p> <p><b>entityID:</b> ND_COUNT</p> <p><b>entityValue:</b> A user-declared variable</p> <p>See example code in the next paragraph.</p>	 <p>“TASK ID” in connects to “TASK ID” out of step 4.</p>

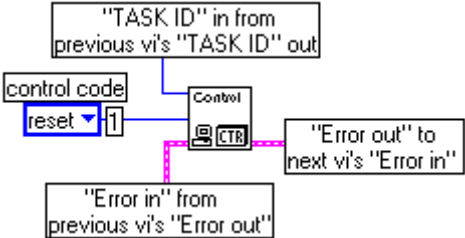
### Example Code for GPCTR Watch

```
u32 ulCount = 0;
```

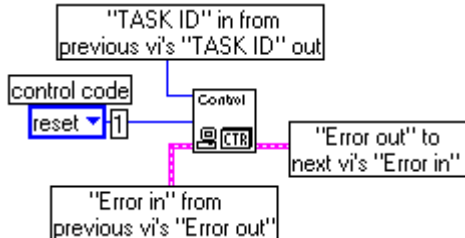
```
iStatus = GPCTR_WATCH (deviceNumber, ND_COUNTER_0, ND_COUNT, &ulCount);
```

```
ulCount = ulCount * 20;
```

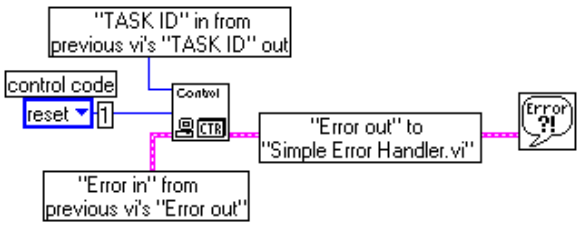
12. Reset counter 0 to its original state.

NI-DAQ Function Call	LabVIEW Block Diagram
<p>Call GPCTR_Control with the following parameters:</p> <p><b>deviceNumber:</b> The value set by MAX</p> <p><b>gpctrNum:</b> ND_COUNTER_0</p> <p><b>action:</b> ND_RESET</p>	

13. Reset counter 1 to its original state.

NI-DAQ Function Call	LabVIEW Block Diagram
<p>Call GPCTR_Control with the following parameters:</p> <p><b>deviceNumber:</b> The value set by MAX</p> <p><b>gpctrNum:</b> ND_COUNTER_1</p> <p><b>action:</b> ND_RESET</p>	 <p>“TASK ID” in connects to “TASK ID” out of step 10.</p>

14. Reset counter 1 on the PXI-6608 to its original state, if necessary.

NI-DAQ Function Call	LabVIEW Block Diagram
<p>Call <code>GPCTR_Control</code> with the following parameters:</p> <p><b>deviceNumber:</b> The value set by MAX</p> <p><b>gpctrNum:</b> <code>ND_COUNTER_1</code></p> <p><b>action:</b> <code>ND_RESET</code></p>	 <p>"TASK ID" in connects to "TASK ID" out of step 3e in the <i>Setting Up the Stable Clock</i> section.</p>

You have now completed measuring the frequency of the crystal oscillator on your 6601/6602 device.

## Comparing the Measured Frequency to 6601/6602 Device Specifications

Compare your calculated frequency from step 11 in the previous section with the following values:

- If the frequency you measure falls between the upper limit of 20,001,500 Hz and the lower limit of 19,998,500 Hz, your device is operating within its specifications.
- If the frequency you measure is above 20,001,500 Hz, your device is nonfunctional. Return the device to National Instruments for repair or replacement.
- If the frequency you measure is below 19,998,500 Hz, your device is nonfunctional. Return the device to National Instruments for repair or replacement.

You have completed verifying the operation of your National Instruments 6601/6602 device.