

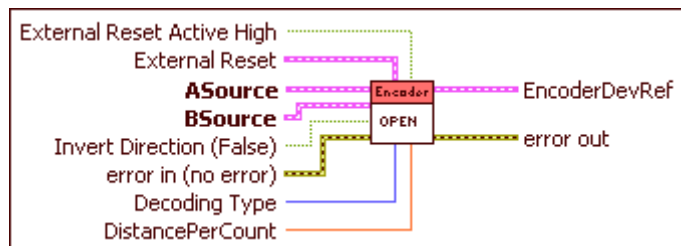
Open VI

Owning Palette: [Encoder](#)

Requires: FIRST Robotics Competition software

Opens a reference to the encoder you specify. You must open a reference before using any other VIs on this palette.

Use the [ToDigitalSource](#) VI to create a digital source that you can wire to the **ASource** and **BSource** inputs of this VI. After you open a reference to the encoder, you can use the [Start](#) and [Stop](#) VIs to start and stop the encoder, respectively.



TF **External Reset Active High** specifies whether the encoder resets when the External Reset digital source is high or low. When **External Reset Active High** is FALSE, the encoder resets when the digital source is low. When **External Reset Active High** is TRUE, the encoder resets when the digital source is high. The default is TRUE.

FF **External Reset** configures the digital source for the external signal that resets the encoder.

FF **DevStatus** describes the error status before this VI or function runs. The default is no error.

TF **status** is TRUE (X) if an error occurred before this node ran or FALSE (checkmark) to indicate a warning or that no error occurred before this node ran. The default is FALSE.

I32 **code** is the error or warning code. The default is 0. If **status** is TRUE, **code** is an [error code](#). If **status** is FALSE, **code** is 0 or a warning code.

abc **source** specifies the origin of the error or warning and is, in most cases, the name of the node that produced the error or warning. The default is an empty string.

TF **AnalogTriggerMode?** specifies, when TRUE, to use an analog trigger output instead of a digital input as the digital source.

20a **DigitalMode** specifies information about the digital input you want to use as the digital source.

20a **DIO Module** specifies the slot number on the CompactRIO device of the digital module you want to use. **DIO Module** can specify a value of **Slot 4** or **Slot 6**. If **DIO Module** specifies a value of **Default**, this VI uses the default digital module. The default digital module is the first digital module, or the module in slot 4.

20a **DIO Channel** specifies the channel of the **DIO Module** that you want to use. **DIO Channel** can specify a value between **DIO 1** and **DIO 14**. If **DIO Channel** is **Disabled**, the FPGA on the CompactRIO device disables the routing destination.


20a **AnalogTriggerMode** specifies information about the analog trigger output you want to use as the digital source.

20a **AnalogTriggerIndex** specifies the index of the reserved analog trigger. **AnalogTriggerIndex** can specify a value between **Trig 0** and **Trig 7**. If **AnalogTriggerIndex** is **Invalid**, this VI returns an error.


20a **OutputType** specifies the output of the analog trigger that you want to use as the digital source.


| | |
|-------------------------|---|
| InWindow (0) | Specifies to use the In Window value of the analog trigger as the analog trigger output. You can use the GetOutput VI to determine the In Window value. If the signal is within the window, the analog trigger output is TRUE. If the signal is outside the window, the analog trigger output is FALSE. |
| TriggerState (1) | Specifies to use the state of the analog trigger as the analog trigger output. The analog trigger output is TRUE if the signal passes between a value within the hysteresis thresholds and a value above the upper hysteresis threshold. The analog trigger output is FALSE if the signal passes between a value within the hysteresis thresholds and a value below the lower hysteresis threshold. |
| Rising (2) | Specifies to use a pulse corresponding to a rising signal as the analog |


| | |
|--------------------|--|
| | trigger output. |
| Falling (3) | Specifies to use a pulse corresponding to a falling signal as the analog trigger output. |


 **ASource** specifies information about the first digital source of the encoder. You can use either a digital input or an analog trigger output as the digital source. Use the [ToDigitalSource](#) VI to create this digital source.

 **DevStatus** describes the error status before this VI or function runs. The default is no error.


 **status** is TRUE (X) if an error occurred before this node ran or FALSE (checkmark) to indicate a warning or that no error occurred before this node ran. The default is FALSE.

 **code** is the error or warning code. The default is 0. If **status** is TRUE, **code** is an [error code](#). If **status** is FALSE, **code** is 0 or a warning code.


 **source** specifies the origin of the error or warning and is, in most cases, the name of the node that produced the error or warning. The default is an empty string.


 **AnalogTriggerMode?** specifies, when TRUE, to use an analog trigger output instead of a digital input as the digital source.


 **DigitalMode** specifies information about the digital input you want to use as the digital source.

 **DIO Module** specifies the slot number on the CompactRIO device of the digital module you want to use. **DIO Module** can specify a value of **Slot 4** or **Slot 6**. If **DIO Module** specifies a value of **Default**, this VI uses the default digital module. The default digital module is the first digital module, or the module in slot 4.

 **DIO Channel** specifies the channel of the **DIO Module** that you want to use. **DIO Channel** can specify a value between **DIO 1** and **DIO 14**. If **DIO Channel** is **Disabled**, the FPGA on the CompactRIO device disables the routing destination.

 **AnalogTriggerMode** specifies information about the analog trigger output you want to use as the digital source.


 **AnalogTriggerIndex** specifies the index of the reserved analog trigger. **AnalogTriggerIndex** can specify a value between **Trig 0** and **Trig 7**. If **AnalogTriggerIndex** is **Invalid**, this VI returns an error.

 **OutputType** specifies the output of the analog trigger that you want to use as the digital source.


| | |
|-------------------------|---|
| InWindow (0) | Specifies to use the In Window value of the analog trigger as the analog trigger output. You can use the GetOutput VI to determine the In Window value. If the signal is within the window, the analog trigger output is TRUE. If the signal is outside the window, the analog trigger output is FALSE. |
| TriggerState (1) | Specifies to use the state of the analog trigger as the analog trigger output. The analog trigger output is TRUE if the signal passes between a value within the hysteresis thresholds and a value above the upper hysteresis threshold. The analog trigger output is FALSE if the signal passes between a value within the hysteresis thresholds and a value below the lower hysteresis threshold. |
| Rising (2) | Specifies to use a pulse corresponding to a rising signal as the analog trigger output. |
| Falling (3) | Specifies to use a pulse corresponding to a falling signal as the analog trigger output. |

 **BSource** specifies information about the second digital source of the encoder. You can use either a digital input or an analog trigger output as the digital source. Use the [ToDigitalSource](#) VI to create this digital source.

 **DevStatus** describes the error status before this VI or function runs. The default is no error.

 **status** is TRUE (X) if an error occurred before this node ran or FALSE (checkmark) to indicate a warning or that no error occurred before this node ran. The default is FALSE.


 **code** is the error or warning code. The default is 0. If **status** is TRUE, **code** is an [error code](#). If **status** is FALSE, **code** is 0 or a warning code.

 **source** specifies the origin of the error or warning and is, in most cases, the name of the node that produced the error or warning. The default is an empty string.

 **AnalogTriggerMode?** specifies, when TRUE, to use an analog trigger output instead of a digital input


 as the digital source.


 **DigitalMode** specifies information about the digital input you want to use as the digital source.

 **DIO Module** specifies the slot number on the CompactRIO device of the digital module you want to use. **DIO Module** can specify a value of **Slot 4** or **Slot 6**. If **DIO Module** specifies a value of **Default**, this VI uses the default digital module. The default digital module is the first digital module, or the module in slot 4.


 **DIO Channel** specifies the channel of the **DIO Module** that you want to use. **DIO Channel** can specify a value between **DIO 1** and **DIO 14**. If **DIO Channel** is **Disabled**, the FPGA on the CompactRIO device disables the routing destination.


 **AnalogTriggerMode** specifies information about the analog trigger output you want to use as the digital source.


 **AnalogTriggerIndex** specifies the index of the reserved analog trigger. **AnalogTriggerIndex** can specify a value between **Trig 0** and **Trig 7**. If **AnalogTriggerIndex** is **Invalid**, this VI returns an error.

 **OutputType** specifies the output of the analog trigger that you want to use as the digital source.

| | |
|-------------------------|---|
| InWindow (0) | Specifies to use the In Window value of the analog trigger as the analog trigger output. You can use the GetOutput VI to determine the In Window value. If the signal is within the window, the analog trigger output is TRUE. If the signal is outside the window, the analog trigger output is FALSE. |
| TriggerState (1) | Specifies to use the state of the analog trigger as the analog trigger output. The analog trigger output is TRUE if the signal passes between a value within the hysteresis thresholds and a value above the upper hysteresis threshold. The analog trigger output is FALSE if the signal passes between a value within the hysteresis thresholds and a value below the lower hysteresis threshold. |
| Rising (2) | Specifies to use a pulse corresponding to a rising signal as the analog trigger output. |
| Falling (3) | Specifies to use a pulse corresponding to a falling signal as the analog trigger output. |


 **Invert Direction (False)** specifies, when TRUE, that the encoder increases the count when the **ASource** signal leads the **BSource** signal and decreases the count when the **BSource** signal leads the **ASource** signal. **Invert Direction (False)** specifies, when FALSE, that the encoder increases the count when the **BSource** signal leads the **ASource** signal and decreases the count when the **ASource** signal leads the **BSource** signal. The default is FALSE.


 **error in (no error)** describes error conditions that occur before this node runs. The default is **no error**. If an error occurred before this node runs, the node passes the **error in (no error)** value to **error out**. This node runs normally only if no error occurred before this node runs. If an error occurs while this node runs, it runs normally and sets its own error status in **error out**. Use the [Simple Error Handler](#) or [General Error Handler](#) VIs to display the description of the error code. Use [exception control](#) to treat what is normally an error as no error or to treat a warning as an error. Use **error in (no error)** and **error out** to check errors and to specify execution order by wiring **error out** from one node to **error in (no error)** of the next node.

 **status** is TRUE (X) if an error occurred before this node ran or FALSE (checkmark) to indicate a warning or that no error occurred before this node ran. The default is FALSE.

 **code** is the error or warning code. The default is 0. If **status** is TRUE, **code** is an [error code](#). If **status** is FALSE, **code** is 0 or a warning code.


 **source** specifies the origin of the error or warning and is, in most cases, the name of the node that produced the error or warning. The default is an empty string.

 **Decoding Type** specifies the type of decoding used to decode the quadrature signal. **Decoding Type** supports three types of decoding for quadrature encoders: 1x, 2x, and 4x.


 **DistancePerCount** scales the pulses of the encoder into engineering units for the **Distance** parameter of the [Get](#) VI. For example, if the wheel diameter is six inches and the encoder pulses 64 times per rotation of the wheel, then a scaling factor of $6\pi / 64 = .2945$ returns the number of inches the wheel has turned. In this case, you input .2945.


 **EncoderDevRef** returns a reference to the encoder.


 **DevStatus** describes the error status that this VI or function produces.

 **status** is TRUE (X) if an error occurred before this node ran or FALSE (checkmark) to indicate a warning or that no error occurred before this node ran.

 **code** is the error or warning code. If **status** is TRUE, **code** is an [error code](#). If **status** is FALSE, **code** is 0 or a warning code.


 **source** specifies the origin of the error or warning and is, in most cases, the name of the node that produced the error or warning.


 **EncoderIndex** returns the index of the reserved encoder. **EncoderIndex** can return a value between **Enc 0** and **Enc 3**. If **EncoderIndex** returns a value of **Invalid**, the VI did not find the encoder you specified, and this VI returns an error.

 **CounterIndex** returns the index of the reserved counter. **CounterIndex** can return a value between **Ctr 0** and **Ctr 7**. If **CounterIndex** returns a value of **Invalid**, the VI did not find the counter you specified, and this VI returns an error.

 **DistancePerCount** returns the pulses of the encoder as engineering units.

 **Decoding Type** returns the type of decoding used to decode the quadrature signal.

 **error out** contains error information. If **error in** indicates that an error occurred before this VI or function ran, **error out** contains the same error information. Otherwise, it describes the error status that this VI or function produces. Right-click the **error out** front panel indicator and select **Explain Error** from the shortcut menu for more information about the error.

 **status** is TRUE (X) if an error occurred before this node ran or FALSE (checkmark) to indicate a warning or that no error occurred before this node ran.

 **code** is the error or warning code. If **status** is TRUE, **code** is an [error code](#). If **status** is FALSE, **code** is 0 or a warning code.

 **source** specifies the origin of the error or warning and is, in most cases, the name of the node that produced the error or warning.

[Submit feedback on this topic](#)