

TI C2000 Toolbox eQEP

This document describes eQEP component from TI C2000 Toolbox library.

Short description

eQEP component enables the user to acquire data for machine position and speed sensing. It processes digital signals by sampling the digital inputs of MCU *Enhanced Quadrature Encoder Pulse* peripheral. These types of signals typically come from incremental encoder that is attached to the machine rotor shaft.

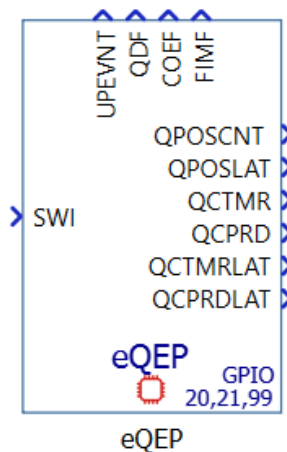


Figure 1. eQEP component icon.

Detailed overview

Component tabs are grouped according to the main eQEP peripheral submodules (see [TI Technical Reference Manual](#)).

NOTE: Machine encoder resolution must be considered when parametrizing the eQEP component!

NOTE: It is recommended to select *target platform* on [TI C2000 Setup](#) component before configuring the component.

NOTE: If [HIL TI LaunchPad Interface](#) is used, connection between the LaunchPad board and the Launchpad Interface should be made via two 5-pin ribbon cables, provided by Typhoon HIL, as explained in the link. On some devices, appropriate eQEP signal routing **must** be configured! Details on how to do this can be found [here](#).

Component properties:

Property labels contain names of relevant bitfields/registers to which properties are mapped, for detailed explanation see [TI Technical Reference Manual](#).

- **Tab General:**

- Module – select the number eQEP module,
- Execution rate – Desired rate at which component inputs will be applied and outputs will be updated. This value must be compatible with other components of the same subsystem: the value must be a multiple of the fastest execution rate in the circuit. To specify the execution rate, you can use either decimal (e.g. 0.001) or exponential values (e.g. 1e-3) in seconds. Alternatively, you can type in 'inherit' in which case the component will be assigned execution rate based on the execution rate of the components it is receiving input from.

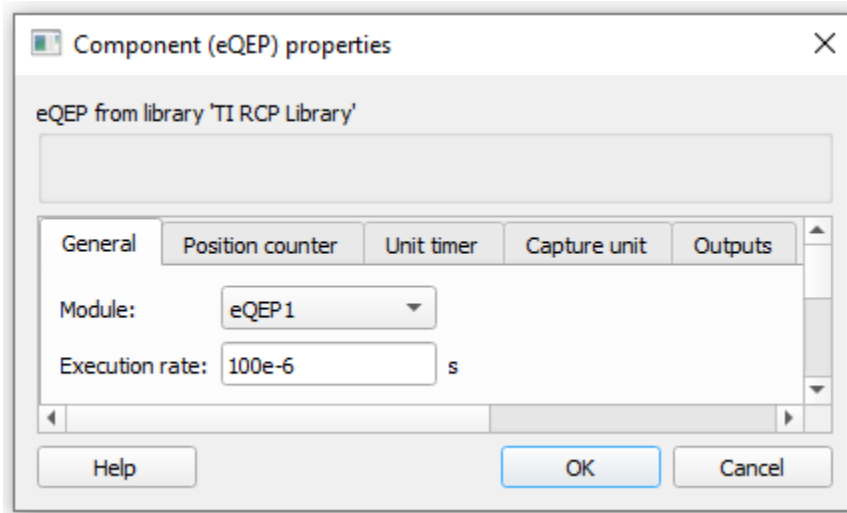


Figure 2. eQEP component dialog - General tab.

- **Tab Position counter:**

- Position counter enable (QPEN) – enables/disables the position counter,
- Signal Source for Position Counter (SWAP) – defines whether the inputs for A and B position counter channels are swapped. **NOTE:** HiL device generates A and B encoder signals that are swapped, hence this bit should be set.
- Position counter reset (PCRM) – specifies an event which will reset the value of position counter register (typically an index signal or timeout),
- External Clock Rate (XCR) – specifies if enhanced position counter resolution is used by counting both rising and falling edges,
- QEP_A input polarity (QAP) – specifies whether the A input polarity is inverted,
- QEP_B input polarity (QBP) – specifies whether the B input polarity is inverted,
- QEP_I input polarity (QIP) – specifies whether the I (index) input polarity is inverted.

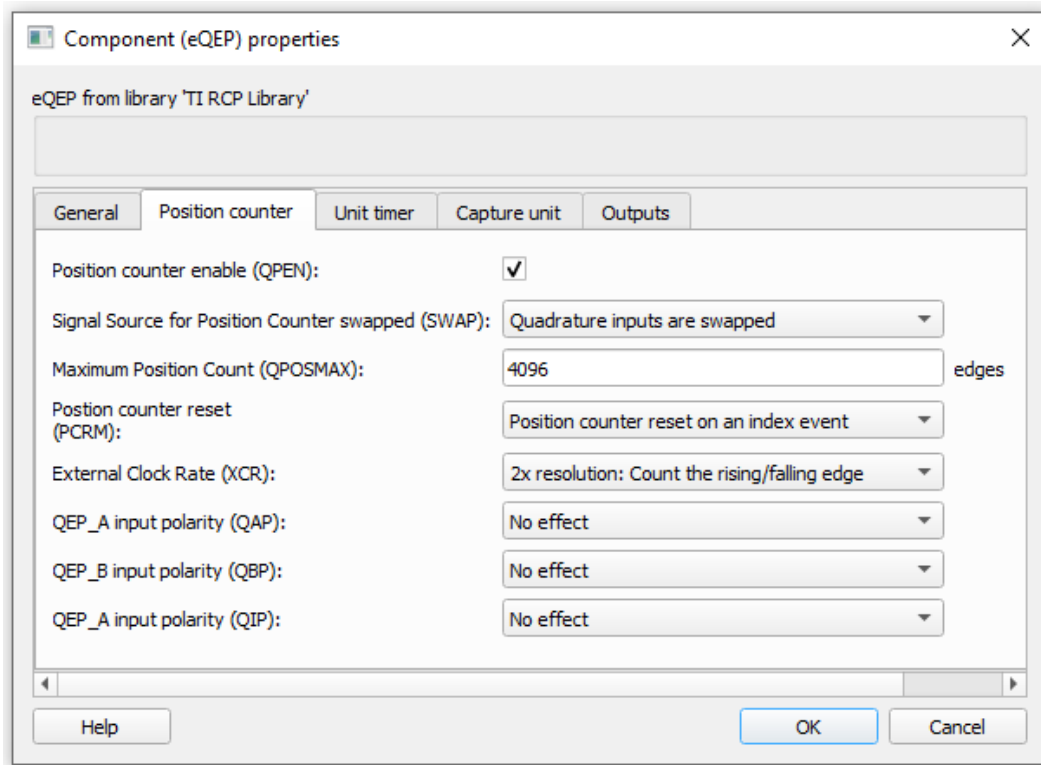


Figure 3. eQEP component dialog - Position counter tab.

- **Tab Unit timer:**
 - Unit timer enabled (UTE) – enables the unit timer sub-module,
 - QEP Unit Period (QUPRD) – period of unit timer in seconds on which the unit timer will generate an event. For example, this event can be used to reset the position counter register.

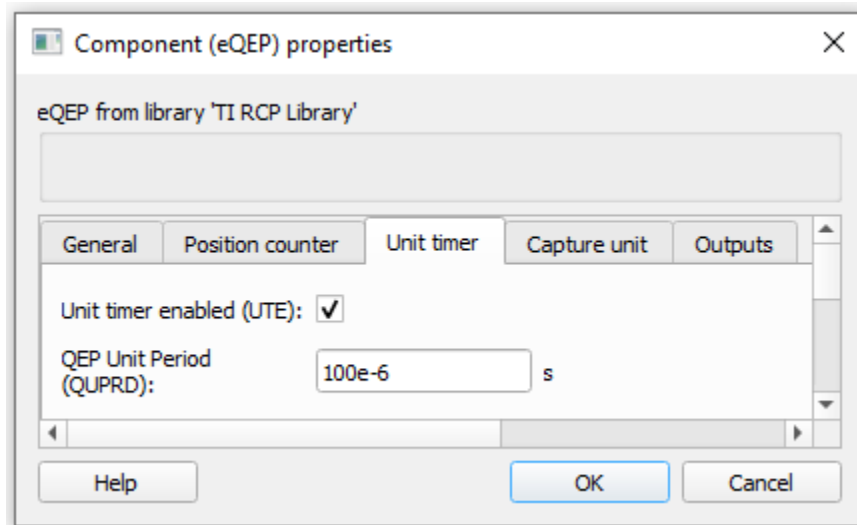


Figure 4. eQEP component dialog - Unit timer tab.

- **Tab Capture unit:**

- Enabled eQEP capture – enables the Quadrature Capture sub-module,
- Capture latch mode (QCLM) – specifies an event when the capture timer (QCTMR) and capture period (QCPRD) values are latched into QCTMRLAT and QCPRDLAT registers,
- eQEP capture timer clock prescaler (CCPS) – clock prescaler for capture unit timer,
- Unit position event (UPEVNT) prescaler (UPPS) – edge event prescaler for capture unit. For example, the capture unit can be configured to detect every 2nd edge by using this prescaler.

Prescalers CCPS and UPPS can be useful to achieve more precise position measurement by pre-defining these values based on expected working machine rotor speed. Since quadrature capture timer is 16-bit, the timer register can overflow on low speeds because the time between two successive edges is too long, so the clock prescaler helps in this case. On contrary, if the speed is too high, this timer will work with less precise time measurement, so the event prescaler can play a role in this situation.

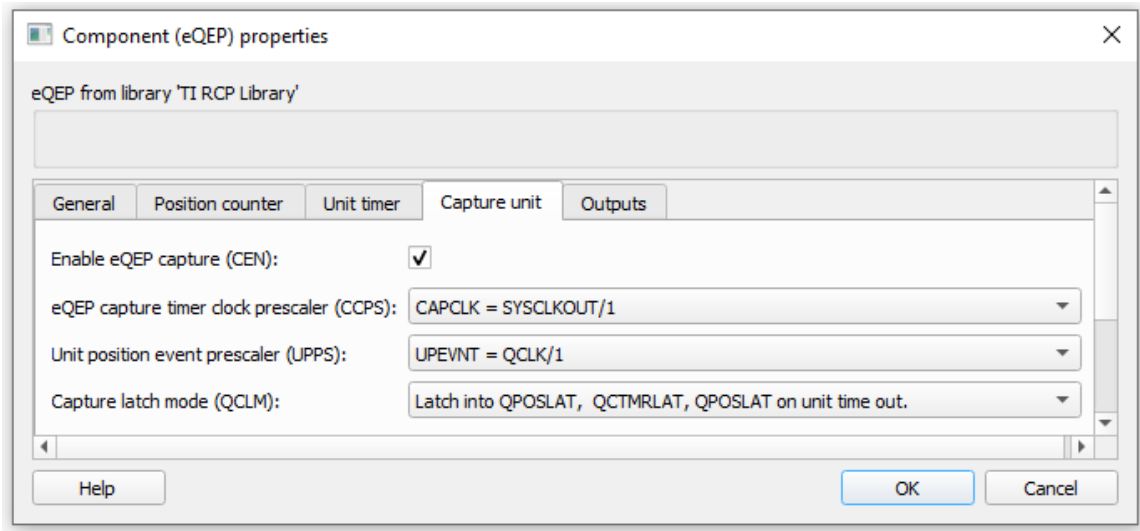


Figure 5. eQEP component dialog - Capture unit tab.

- **Tab Outputs**

- Each property is a checkbox which indicates whether the register value is read. For every checked output, separate output port is created on the component and the value of the corresponding register/bitfield is applied.
- Values stored into selected (checked) registers from this tab are applied to component outputs on execution rate. Counter and latched counter register values are applied to terminals on the right side, while selected

flag (status) register values are applied to terminal on the top side of the component icon.

Ports:

Port labels contain names of relevant bitfields/registers to which properties are mapped, for detailed explanation see [TI Technical Reference Manual](#).

- **Outputs**

- QPOSCNT – position counter register – number of counted events,
- QPOSLAT – position counter register value latched on a unit time out event,
- QCTMR – timer value for edge capture unit,
- QCPRD – capture period value between the last successive edge events,
- QCTMRLAT – capture timer value latched on event defined by *QCLM* property,
- QCPRDLAT – capture period value latched on event defined by *QCLM* property,
- UPEVNT – edge event flag, indicates whether a edge event is detected,
- QDF – direction flag – 0 for CCW rotation and 1 for CW rotation,
- COEF – capture timer overflow error flag,
- FIMF – first index marker flag, it is set by the first occurrence if index pulse. Once this flag has been set, if the flag is cleared the flag will not be set again until the module is reset by a peripheral or system reset.

For **all** outputs:

- Supported type: real
- Vector support: no

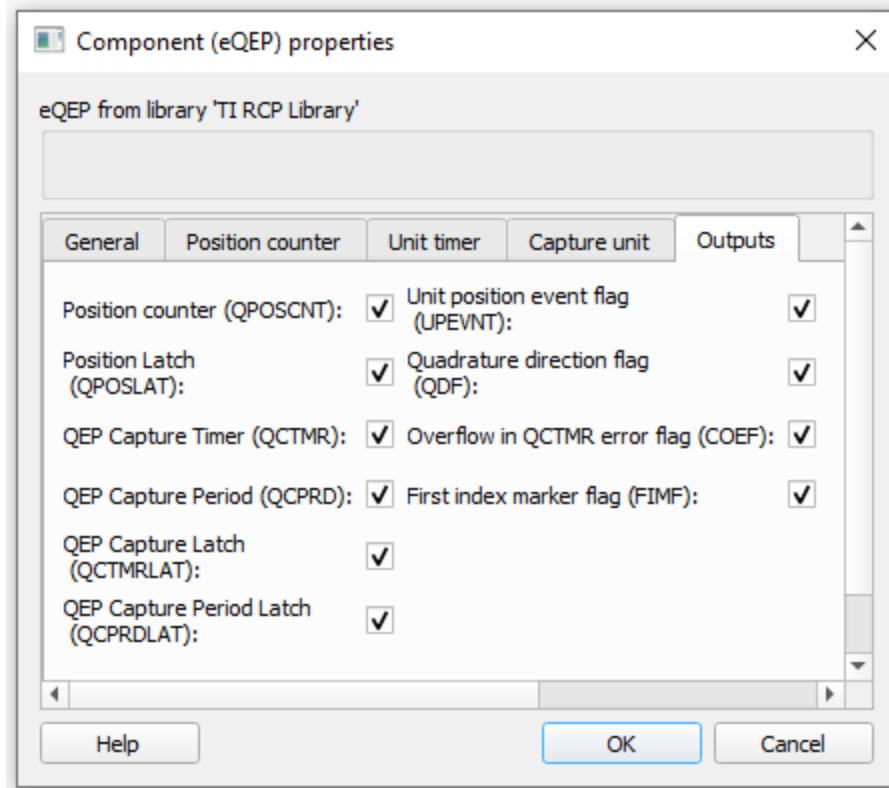


Figure 6. eQEP component dialog - Outputs tab.

- **Inputs**

- SWI – Software position counter initialization. A signal with rising edge aligned with the simulation start should be connected to this input (most commonly, a digital output of the HIL device that is always in high state during the simulation will change its state from low to high on simulation start). This signal resets the value of the position counter register (*QPOSCNT*). The reason behind this is the fact that motor position resets before simulation starts, but the DSP is not aware of that so the old, non-zero position counter register value does not correspond to the rotor position anymore which can cause unexpected behavior.
 - Supported type: uint
 - Vector support: no