User's Guide
SNLU222B–December 2016–Revised September 2019

# FPC401 and FPC402 Evaluation Module (EVM) User's Guide



**Contents**



## Trademarks

All trademarks are the property of their respective owners.

# 1 Hardware Description and Setup

The general procedure for setting up the FPC401 and FPC402 Evaluation Modules (EVM) is described in the following sections.

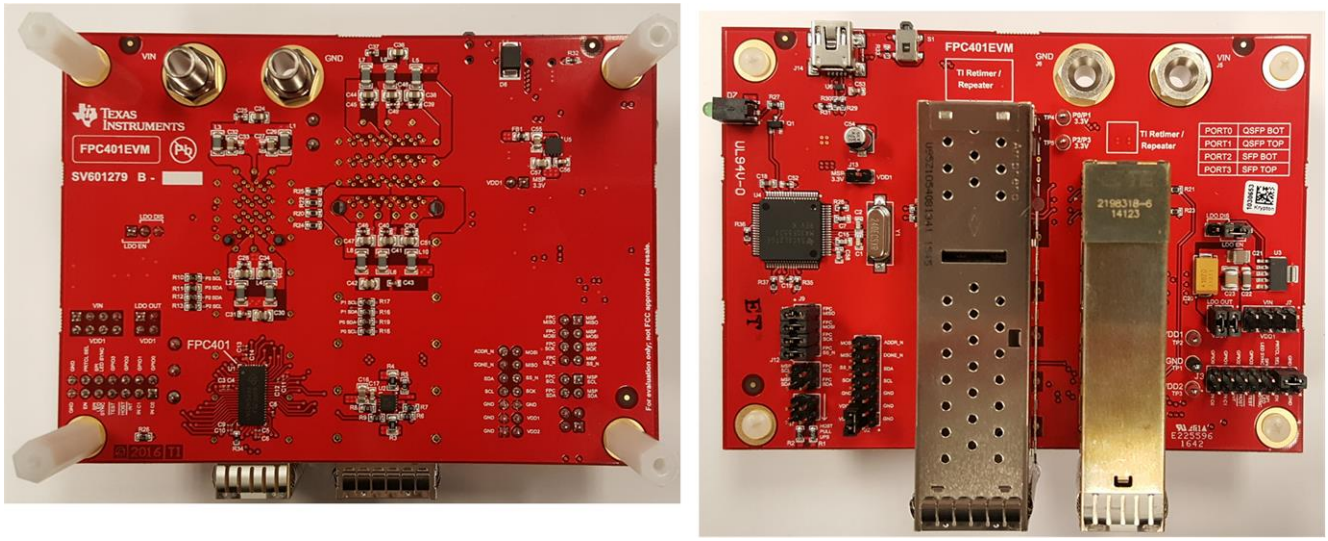


**Figure 1. FPC40xEVM – Top and Bottom**

## 1.1 Power Configuration

Configure the power illustrated by Figure 2.

1. If the LDO is used to power the FPC, enable the LDO through header J8 and connect the LDO output to VDD1 using header J4 (note that each header connection is rated for 1 A of current). Make sure headers J7 and J13 are disconnected. The LDO has a maximum rated current of 1.5 A, so if the cumulative QSFP/SFP module current draw is greater than that, the LDO may not be used.
2. If the LDO is not used and the FPC is powered through an external supply, configure the J8 header to disable the LDO and connect VIN to VDD1 using header J7 (note that each header connection is rated for 1 A of current). Make sure headers J4 and J13 are disconnected.
3. If USB power is used to power the FPC, ensure the LDO and VIN and disconnected from VDD1 (disconnect headers J4 and J7). Connect the 3.3 V of the MSP430 microcontroller to VDD1 through header J13. Note that a typical computer USB port is only able to supply up to 100 mA which will be enough to power the FPC and passive modules (that is, DAC); but it will not be enough to power active SFP+ and QSFP+ modules which are plugged in.
4. To set VDD2 to 3.3 V, connect it to VDD1 through header J2. To operate at lower host-side supply voltages (down to 1.8 V), connect an external supply to the header or test point loop (TP3).
5. The FPC401EVM and FPC402EVM includes a TPS2561 power switch which is used to selectively enable and disable 3.3-V power to the SFP and QSFP ports. The FPC controls this power switch with its GPIO pins. To supply power to ports 0 and 1 (QSFP+ ports and LEDs), configure the FPC GPIO0 high logic output to enable the power switch. Similarly, configure GPIO1 to high logic to enable power for ports 2 and 3 (SFP+ ports and LEDs). Alternatively, supply 3.3 V externally through the test points (TP4 and TP5). Note that the ground is shared with the board.
6. GPIO2 and GPIO3 are connected to the fault outputs of the power switch, and hence should be set as inputs or left unconfigured.

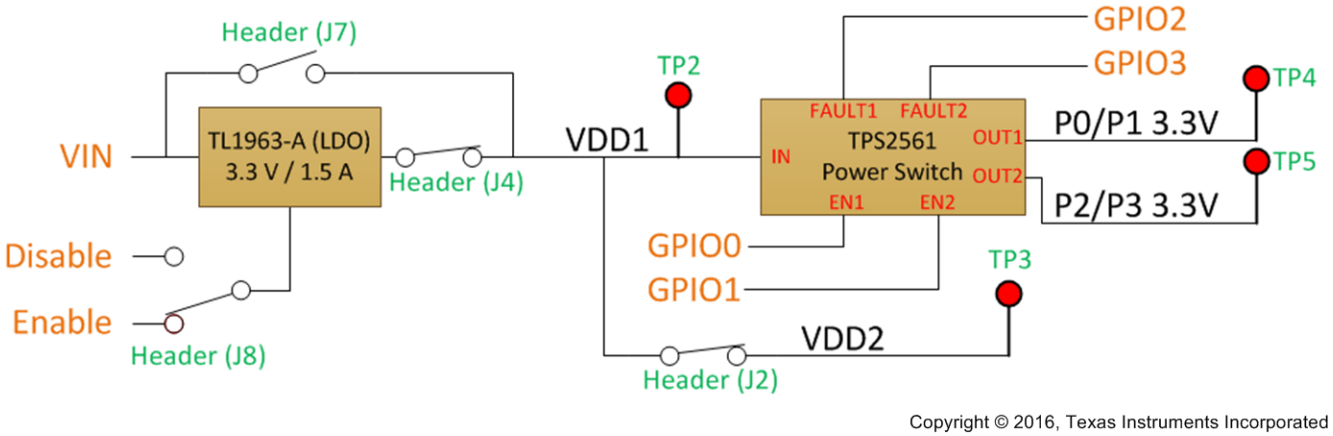


**Figure 2. Power Block Diagram – Configured for LDO Use**

## 1.2 Communication Protocol Configuration

The following sections describe how to configure the FPC401EVM and FPC402EVM for I2C and SPI protocols.

### 1.2.1 I2C Configuration

For I2C communication with the onboard MSP430, refer to Figure 3 and configure the board based on the following procedure.

1. Leave the PRTCL_SEL pin disconnected (header J3).
2. Connect the MSP430 SCL and SDA to the FPC SCL and SDA through header J12. The SPI header (J9) should be disconnected.
3. Connect the 2.7-kΩ pullup resistors through header J1.
4. Connect the VDD2 pin to the VDD1 pin through header J2.
5. Connect the ADDR_N pin to GND through header J2 with the provided jumper wire.

For I2C communication with an external host controller, disconnect the MSP430 SDA and SCL. Connect the FPC SDA and SCL pins to the SDA and SCL pins of the host controller. Only connect the onboard pullup resistors if the host board does not have pullups.

To control multiple EVMs, connect all the SDA and SCL pins together. Only one board should have the MSP430 and pullup resistors connected to these pins. The ADDR_N of the first device should be tied to GND as shown in Figure 3 . Connect DONE_N pin of the first device to the ADDR_N of the second device, and the DONE_N pin of that device to the following device in a daisy-chained fashion until all ADDR_N pins are connected. Connect the final DONE_N pin to GND. See the data sheet for more details on how I2C is configured.

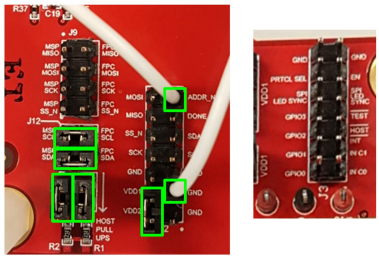

**Figure 3. I2C Configuration**

### 1.2.2 SPI Configuration

For SPI communication with the onboard MSP430, refer to Figure 4 and configure the board based on the following procedure.

1. Connect the PRTCL_SEL pin to GND through header J3.
2. Connect the MSP430 MISO, MOSI, SCK, and SS_N pins to the FPC MISO, MOSI, SCK, and SS_N pins through header J9. The I2C header (J12) should be disconnected.
3. Connect the VDD2 pin to the VDD1 pin through header J2.

For SPI communication with an external host controller, disconnect the MSP430 MOSI, MISO, SCK, and SS_N pins. Connect the FPC MOSI, MISO, SCK, and SS_N pins to the MOSI, MISO, SCK, and SS_N pins of the host controller.

To control multiple EVMs with SPI, connect all the SCK and SS_N pins to a single MSP430 SCK and SS_N pins. Daisy chain the MOSI and MISO pins so that the MOSI of the MSP430 connects to the MOSI of the first device, the MISO of the first device connects to the MOSI of the second device, and so on, until the final device's MISO connects to the MISO of the MSP430. See the data sheet for more details on how SPI is configured.

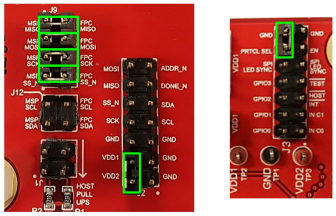


**Figure 4. SPI Configuration**

## 2 FPC401 EVM GUI Description and Setup

The FPC401 EVM GUI is used to program both FPC401EVM and FPC402EVM. The general procedure for setting up the FPC401 GUI is described in the sections below.

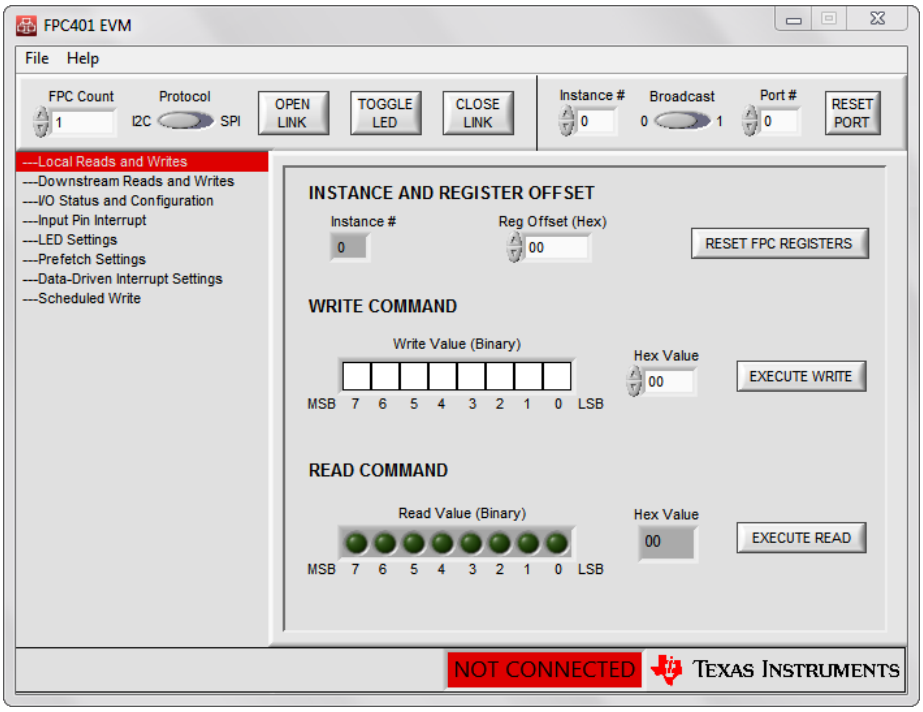


**Figure 5. FPC401 EVM GUI**

## 2.1 Installation

The FPC401 EVM GUI has two installers. The *Setup_FPC401_EVM.exe* installer does not contain the *National Instruments Run-Time Engine* and requires web access to download the NI run-time engine if the computer does not already have this installed. The *Setup_FPC401_EVM_with_RTE.exe* installer is a larger file that contains the run-time engine and does not require web access. Select the appropriate installer and follow the installer wizard to install the FPC401 EVM GUI.

## 2.2 Initializing Communication

To initialize communication with the FPC401EVM, first select the number of FPC401s under *FPC Count* and the protocol (I2C or SPI), highlighted by Figure 6. Click *OPEN LINK* to establish communication with the MSP430 through USB and initialize the device. In I2C mode, this will configure the MSP430 and automatically configure the I2C addresses of all connected FPC401s so that each one has a unique address. In SPI mode, this will configure the MSP430 and establish the length of the SPI message. Each time the hardware configuration changes, the link must be reestablished. For example, connecting additional FPC401s or changing the communication from I2C to SPI requires the link to be reestablished.

The *TOGGLE LED* button will toggle the LED (D7) for a visual indication that the USB communication is working. The text in the bottom of the GUI should say *CONNECTED* and turn green after USB communication is established. Note that this means that the computer is communication with the MSP430 but does not necessarily mean that communication is established with the FPC401.

The *CLOSE LINK* button will terminate USB communication.

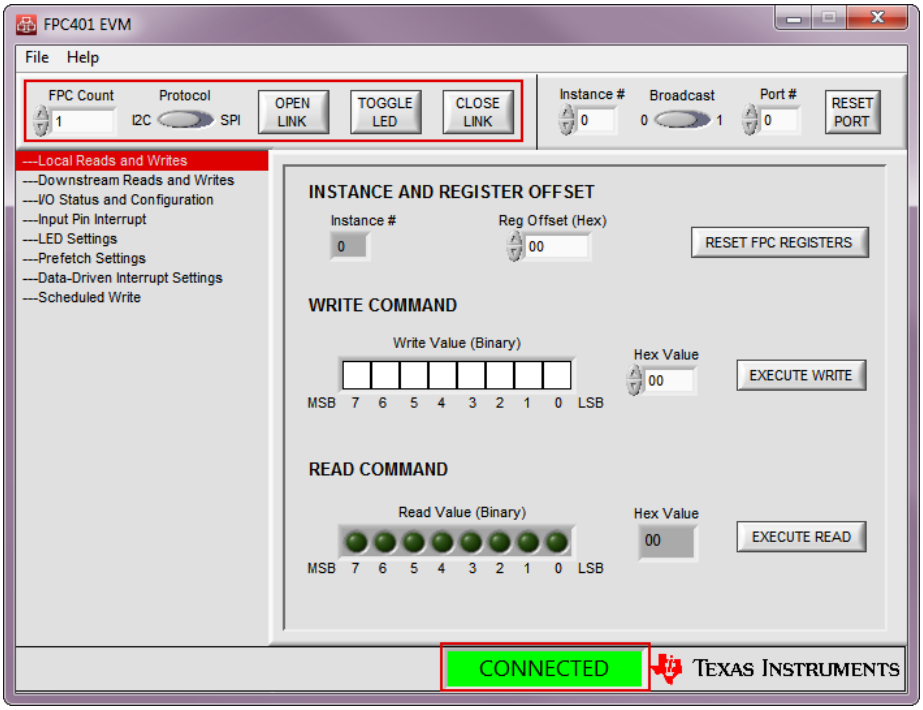


**Figure 6. Communication Control**

The top right section of the GUI highlighted by Figure 7 controls the FPC(s) being programmed by all controls in this GUI. The *Instance #* selects which FPC is being controlled, starting at instance 0. The *Port #* selects which port is being configured, where 0 is the port 0 of FPC instance 0, and 4 is port 0 of FPC instance 1. The *Broadcast* switch will broadcast commands to all FPCs if set to 1. Note that if a command that targets a port is broadcasted, not all ports will be modified. Instead, only ports that match the local port of the selected instance are modified. For example, broadcasting a command that targets port 5 (local port 1 of instance 1) will affect ports 1, 5, 9, 13, and so forth.

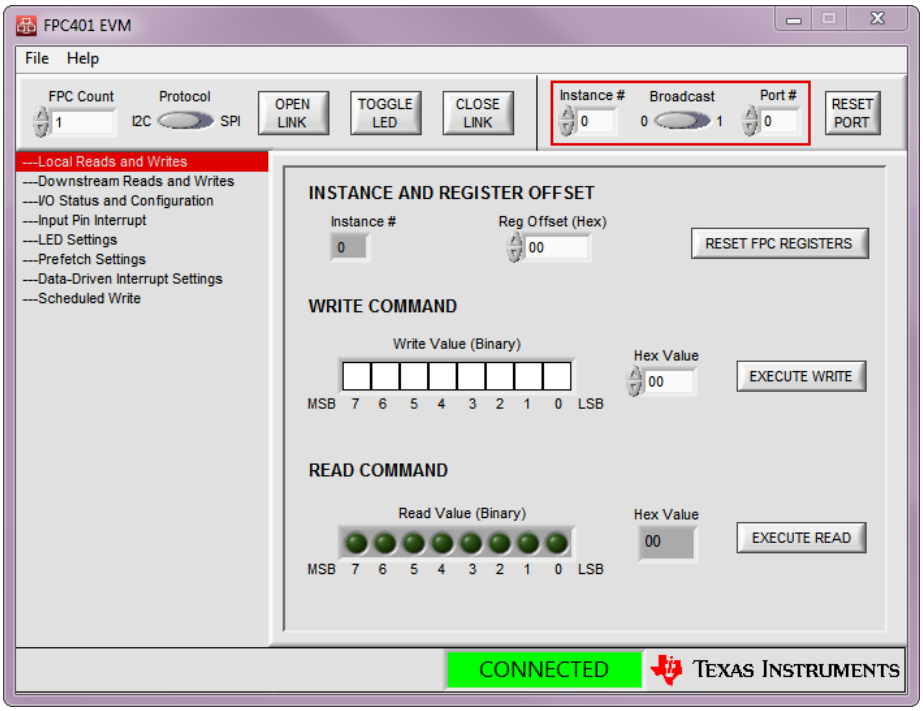


**Figure 7. FPC Instance and Port Selection**

## 2.3 Local FPC Read and Write

To read and write to local FPC registers, select the *Local Reads and Writes* tab. Select the instance number and register offset address to be read or written to. To write, select a value using the radio boxes in the *WRITE COMMAND* section or type the desired hex value and click *EXECUTE WRITE*. To read, click *EXECUTE READ*. The *RESET FPC REGISTERS* button will reset the instance's local FPC registers. Note that performing a register reset in I2C mode will restore the default I2C address of the device and the link must be closed and reopened to restore normal GUI operation.

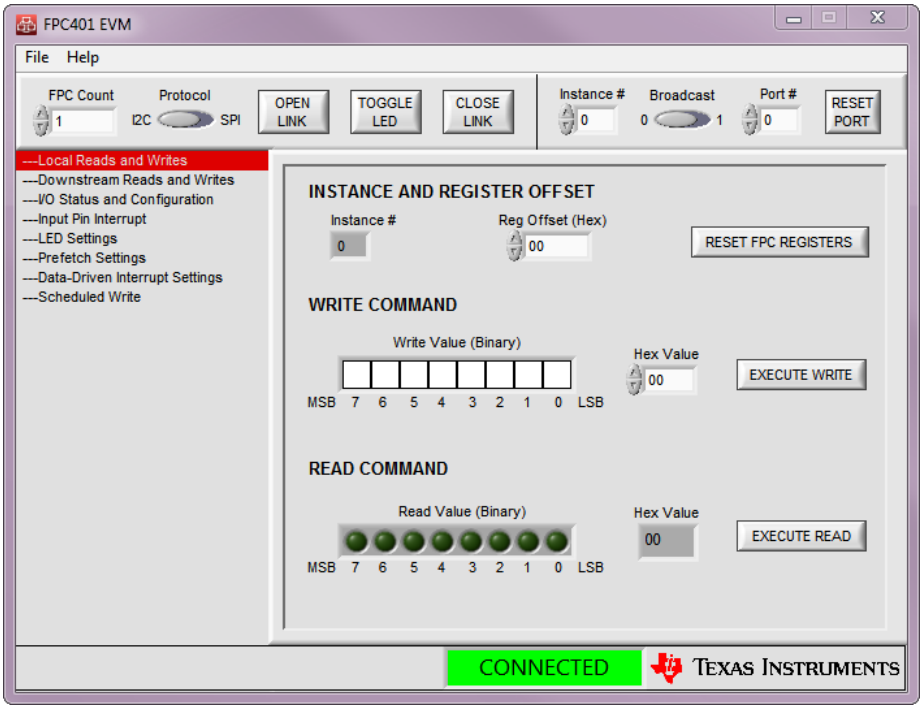


**Figure 8. FPC401 GUI – Local Reads and Writes**

## 2.4 Remote Access on Downstream Modules

To perform a remote access read or write on a downstream port, select the *Downstream Reads and Writes* tab. Select the port number, downstream device I2C address, and register offset address. To write, select a value using the radio boxes in the *WRITE COMMAND* section or type the desired hex value and click *EXECUTE WRITE*. To read, click *EXECUTE READ*. The *Set I2C Rate* switch will toggle the I2C rate of the downstream between 100 kHz (SFP) and 400 kHz (QSFP).

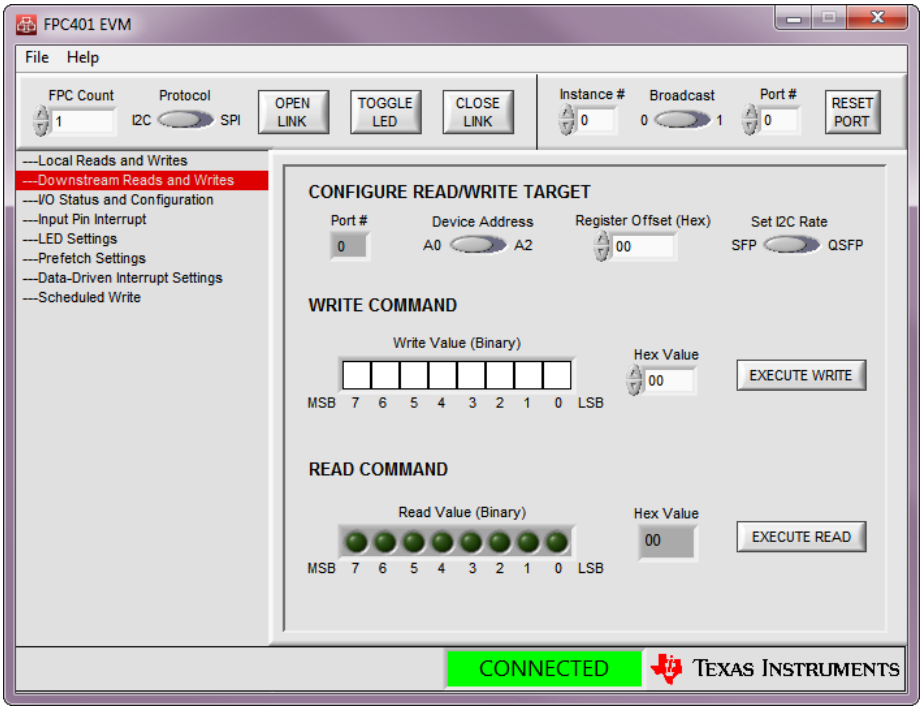


**Figure 9. FPC401 GUI – Downstream Reads and Writes**

## 2.5 Input and Output Status and Configuration

To control the input, output, and GPIO pins, select the *I/O Status and Configuration* tab. The *Enable Periodic Update* button will update the indicator of each input, GPIO, and output periodically set by the *Update Period (ms)* slider. This will also update the *Port's Aggregated Interrupt Flag* which determines if any interrupt of the selected port has been triggered. To modify the state of the GPIOs, modify the *GPIO 0/1/2/3 Set* drop-down menus and click *SET GPIOS*. To modify the state of the outputs, modify the *OUT A/B Set* drop-down menus and click *SET OUTPUTS*.

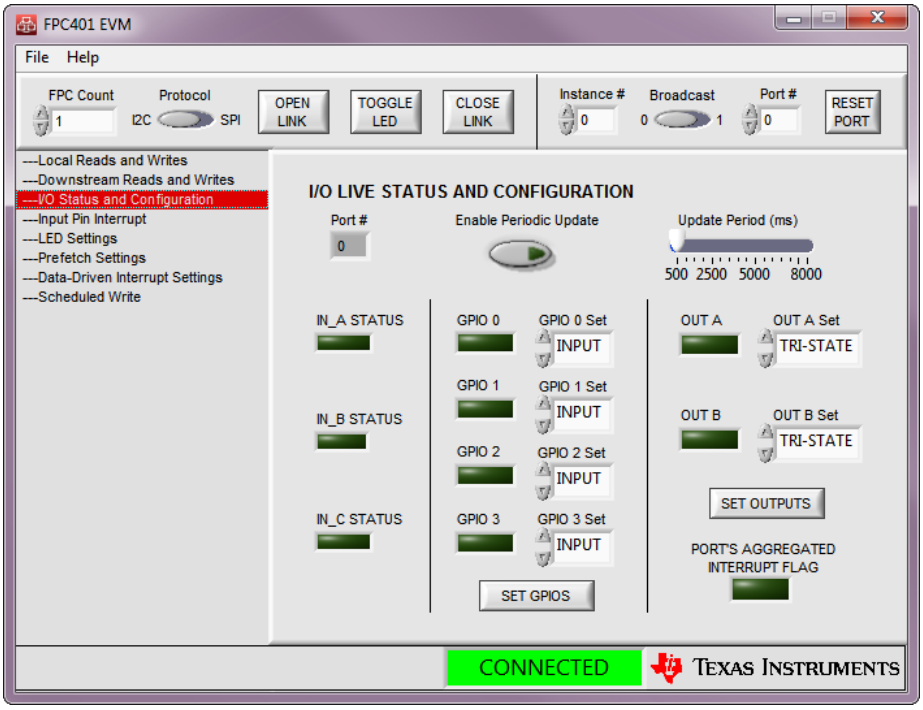


**Figure 10. FPC401 GUI – Input, Output, GPIO Status, and Control**

## 2.6 Interrupt Configuration and Monitoring

To configure the port's interrupts, select the *Input Pin Interrupt* tab. To enable input pin interrupts, select the port, input trigger source, and trigger edge. Click *SET* to configure. To check and clear the status of all of the interrupts of the selected port, click *UPDATE AND CLEAR*. Some of these interrupts are configured in other sections.

To enable the SDA and SCL instances stuck at 0 interrupts, click *SDA STUCK ENABLE* and *SCL STUCK ENABLE* buttons. These buttons enables the interrupts for all ports of the selected instance. To clear these interrupts, a port reset must be issued using the *RESET PORT* button, highlighted in Figure 12.

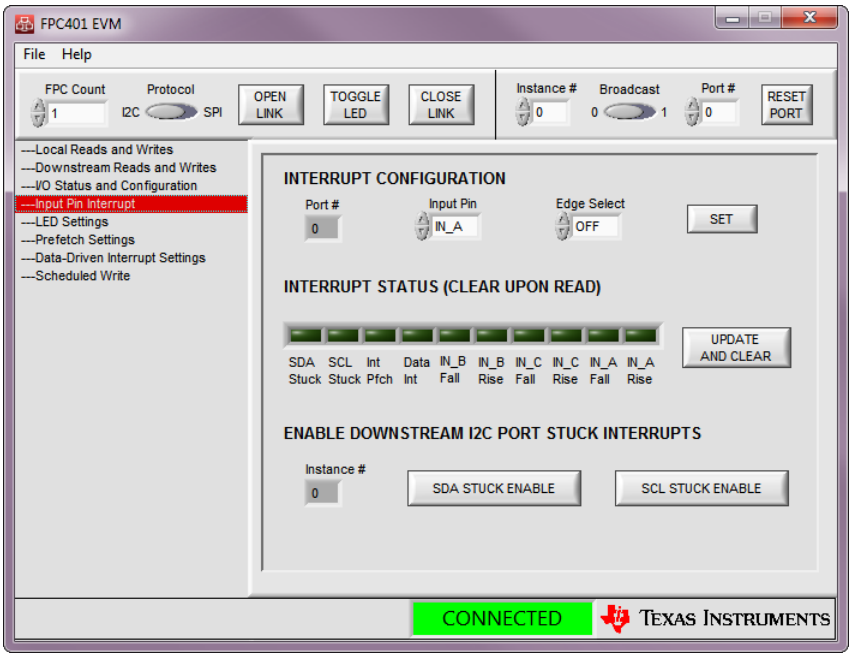


**Figure 11. FPC401 GUI – Interrupt Control**

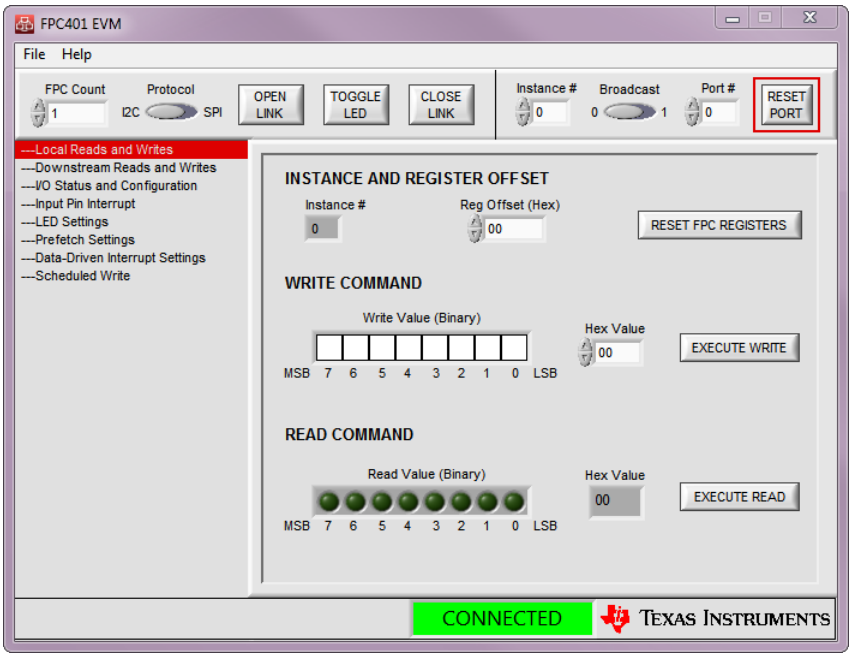


**Figure 12. FPC401 GUI – Port Reset**

## 2.7 LED Configuration

To configure the LEDs, select the *LED Settings* tab. Select the port number, LED number, LED state and click the *SET* button in the *LED STATE* section. If configured in the *PWM* or *BLINK* mode, use the *Duty Cycle (%)* slider and click *SET* in the *PWM MODE SETTINGS* section. Since the PWM duty cycle is 0 by default, the LEDs will not be visible until this is modified. In *BLINK* mode, configure the *On Time* and *Off Time* and click *SET* under the *BLINK MODE SETTINGS* section. To synchronize the LEDs, click the *SYNCHRONIZE LEDS* button at the bottom of the page. This will automatically forward the LED clock for synchronization across multiple devices. See the data sheet for more details.

Note that in the FPC401EVM and FPC402EVM the GPIO0 and GPIO1 must be set to drive high logic to enable the power switch that powers the LEDs and modules.

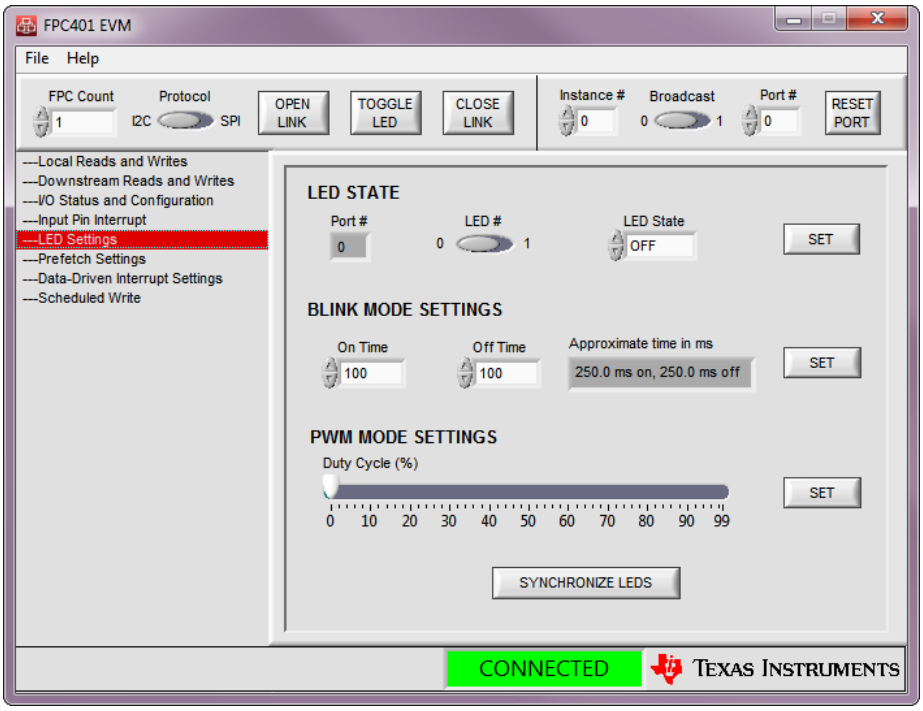


**Figure 13. FPC401 GUI – LED Settings**

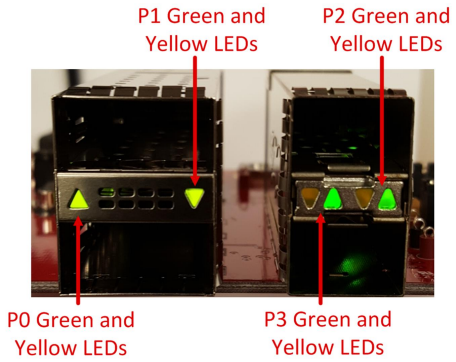


**Figure 14. LED Mapping**

## 2.8 Prefetch Configuration

To configure prefetching, select the *Prefetch Settings* tab. The PREFETCH GLOBAL SETTINGS determines the port number, device I2C address (0xA0 or 0xA2), prefetch length, and starting register offset address for both the periodic prefetch and interrupt-driven prefetch features. The *Prefetch Length* input controls the number of bytes to be prefetched and the *Prefetch Offset (Hex)* input determines the starting register offset address.

In the *PERIODIC PREFETCH SETTINGS* section, enter the prefetch period in milliseconds and click *START PREFETCH*. A period of 0 ms is a one-time prefetch. To stop the periodic prefetch, click the *STOP PREFETCH* button. This will not clear the gate bit, so any downstream read in the prefetched range will read from the FPC memory. To reset the gate bit and force the FPC to read from the downstream port, click *RESET GATE*. See the data sheet and programmer's guide for more details about prefetching.

In the *INTERRUPT DRIVEN PREFETCH* section, select the input that will trigger the interrupt-driven prefetch, the edge transition of the selected input, and click *SET*. After an interrupt-driven prefetch is successful, to clear the gate bit and perform another one, click the *RE-ARM* button. This will also clear the interrupt.

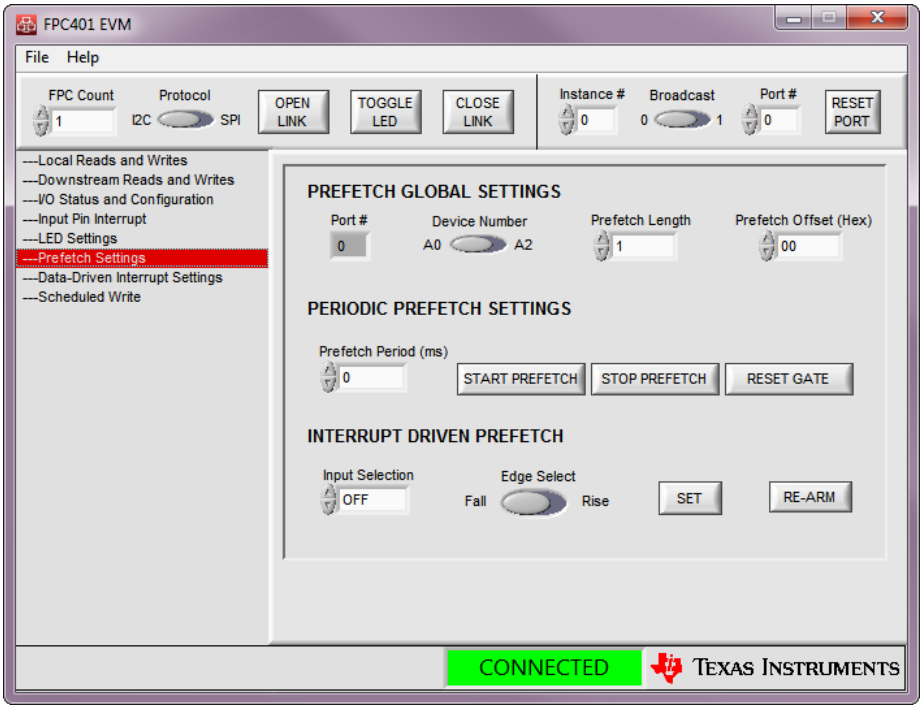


**Figure 15. FPC401 GUI – Prefetch Settings**

## 2.9 Data-Driven Interrupt Configuration

To configure data-driven interrupts, select the *Data-Driven Interrupt Settings* tab. The FPC is able to monitor up to four bytes per port, and the monitored byte index is controlled by the *Byte Number* input. Select the port number, register address to be monitored and the bits to be monitored and click *SET*. Note that the monitored register address must be within the prefetched range for the selected port. Click *CHECK / CLEAR DATA* to determine which bits of the four monitored bytes triggered the interrupt. This will also clear the interrupt.

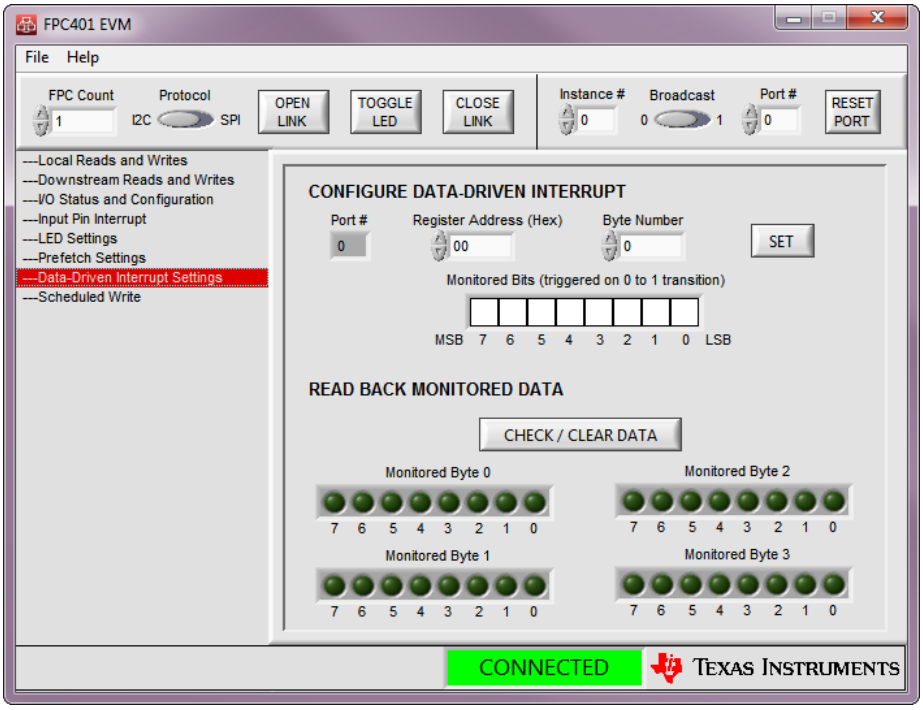


**Figure 16. FPC401 GUI – Data-Driven Interrupt Control**

## 2.10 Scheduled Write Configuration

To perform a scheduled write to one or more downstream ports, select the *Scheduled Write* tab. Select the port number, device address, register offset address, and write value and click *EXECUTE SCHEDULED WRITE*. This will issue a write to the selected port, and may be broadcasted to write to multiple ports across multiple FPC instances. The CONFIGURE COMMON SCHEDULED WRITE section works similarly, but the user may also select which of the ports to be written, allowing writing to all four local ports simultaneously.

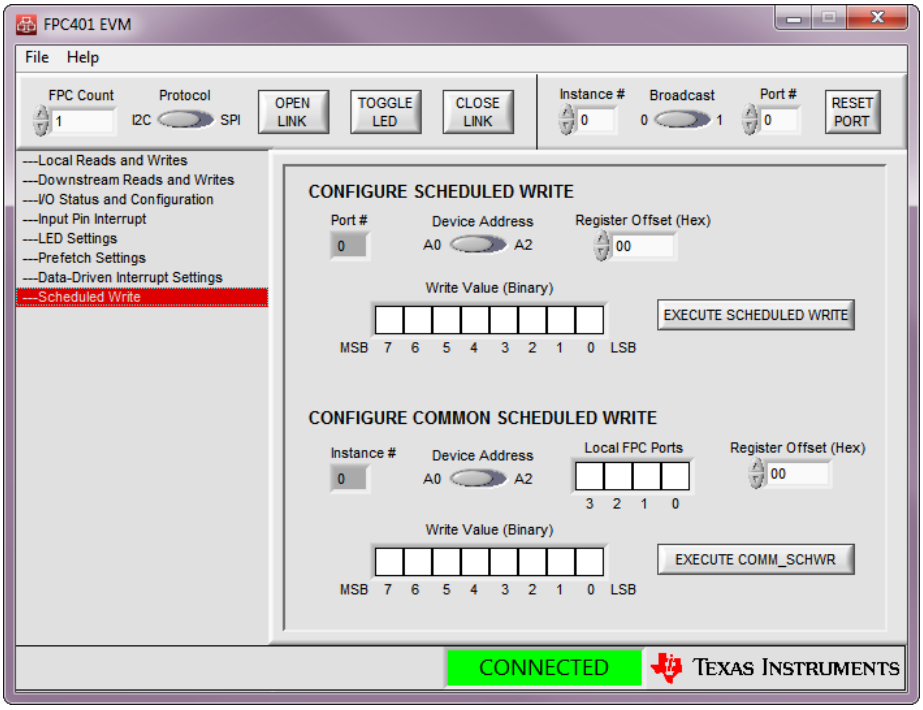


**Figure 17. FPC401 GUI – Scheduled Write Control**

## Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

| Changes from Original (November 2016) to B Revision | Page |
|---|---|
| • First public release. | 2 |

| Changes from Original (November 2016) to B Revision | Page |
|---|---|
| • First public release. | 2 |

# IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265