Introduction

The evaluation board is designed to help the customer evaluate the following devices.

Product Number	Description
9FGL0841	8-output PCIe Clock Generator 3.3V ZOUT = 100Ω
9FGV0841	8-output PCIe Clock Generator 1.8V ZOUT = 100Ω
9FGU0841	8-output PCIe Clock Generator 1.5V ZOUT = 100Ω
9FGL0851	8-output PCIe Clock Generator 3.3V ZOUT = 85Ω

The devices are programmable through SMBus interface. This user guide details the board set and connection as well as the companion GUI installation for communicating to the device. The board has a self contained USB to SMBus interface.

Board Overview

Use the following diagram and table to identify: power supply jacks, USB connector, input and output frequency SMA connectors.

Figure 1. Evaluation Board Overview for the 9FGL0841–100Ω Differential

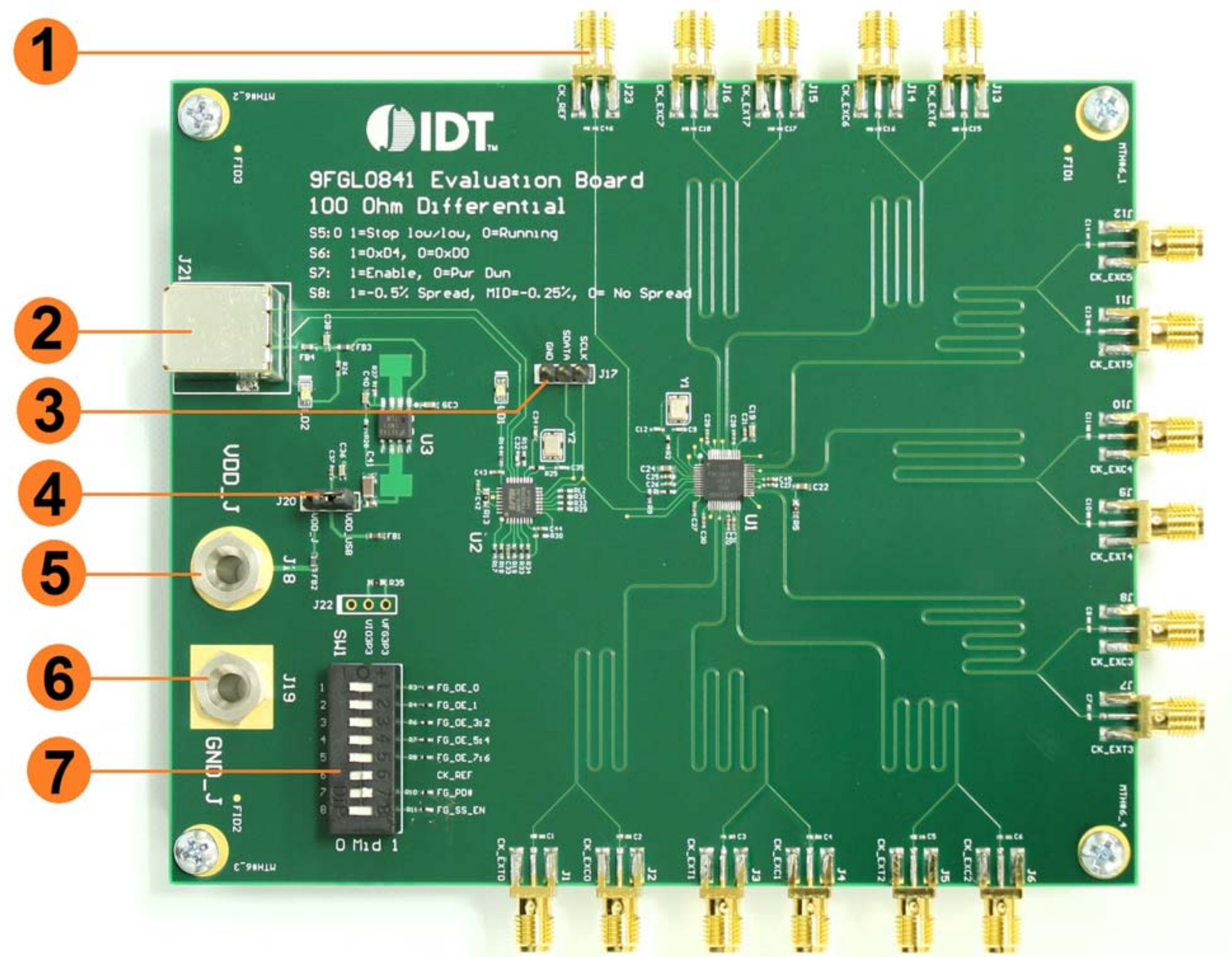


Table 1: EBV Pins and Functions

Item	Name	On-Board Connector Label	Function
1	Outputs 0-7	J1-J16	Low power HCSL outputs
2	USB Interface	J21	Used for connection with a PC and for interaction with the IDT PCIe GUI
3	I^{2}C Connection Port	J17	Used for an external I^{2}C connection
4	Input Voltage Selector	J20	Used for selection of USB power supply or external power supply from J18
5	Power Supply Jack	J18	Input power supply
6	Ground Jack	J19	Used for GND
7	DIP Switch	SW1	S1: FG_OE_0 S2: FG_OE_1 S3: FG_OE_3:2 S4: FG_OE_5:4 S5: FG_OE_7:6 S6: CK_REF S7: FG_PD# S8: FG_SS_EN

Board Power Supply

By default, the board is powered from the USB connector.

Bench Power Supply – An external power supply can be used by connecting jumper J20 between the central pin and the VDD_J position. VDD_J must then be connected to the appropriate power supply for the device ordered.

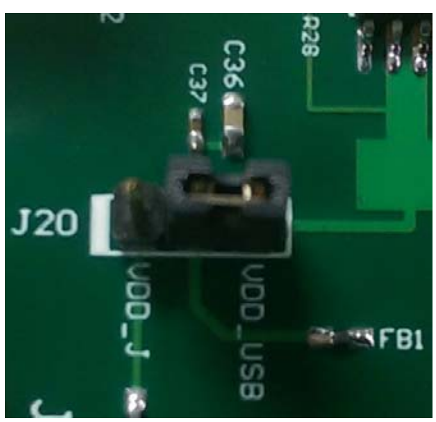
- 9FGL= 3.3V
- 9FGV= 1.8V
- 9FGU = 1.5V

USB Power Supply – When the board is connected to a PC through a USB cable, on-board voltage regulators can supply the appropriate voltage to the clock chip. USB power is selected by connecting J20 between the central pin and the VDD_USB pin.

Depending on the evaluation board ordered, the R22 resistor will be pre-populated as follows:

- For V_{DD} = 1.5V: R22 = 49.9Ω
- For V_{DD} = 1.8V: R22 = 107Ω
- For V_{DD} = 3.3V: R22 = 402Ω

Figure 2. Connecting the jumper to VDD_J or VDD_USB. Default is to power by USB

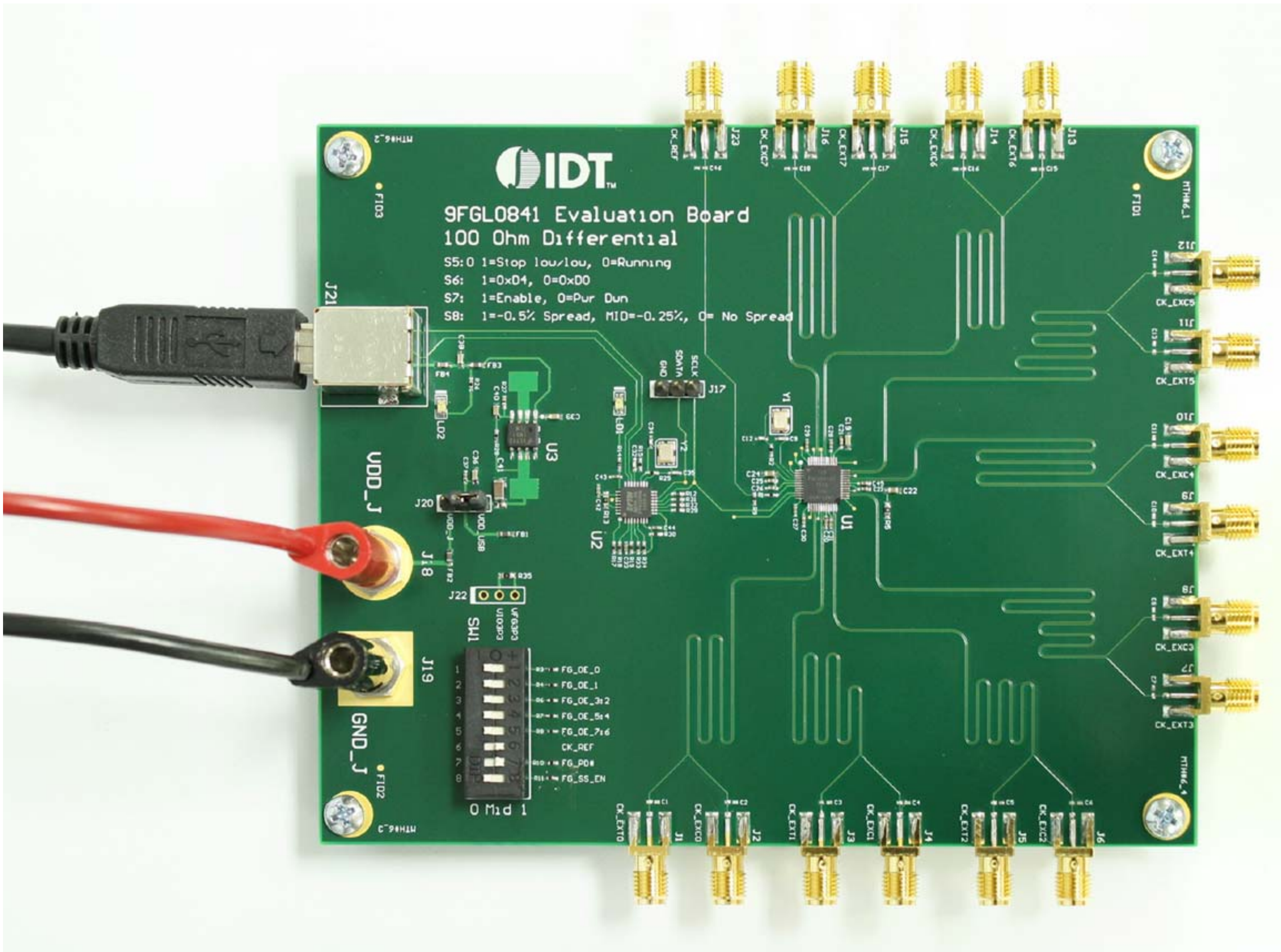


Connecting the Board

The board is connected to a PC through a USB connector for configuring the device, as shown in Figure 3 below. The USB interface will also provide +5V power supply to the board, from which on-board voltage regulators generate various voltages for the core as well as for each output. LED LD2 will light up to indicate a successful connection

The board can also be powered by a bench power supply by connecting one banana jack J18 for the core voltage, respectively. Please see board power supply section for details.

Figure 3. Connecting the Board with USB Port for Communications with Software GUI



PCIe GUI Installation Setup

First the GUI requires a driver for the FTDI IC that interface between the USB and SMBus interfaces.

1. Unzip the files from the PCIe GUI archive on your PC. PCIe GUI zip file can be found at http://www.idt.com/document/swr/software-pcie-evaluation-kits
2. Extract the FTDI windows driver from the PCIe GUI archive or go to the FTDI website to download the latest driver and install on your computer.

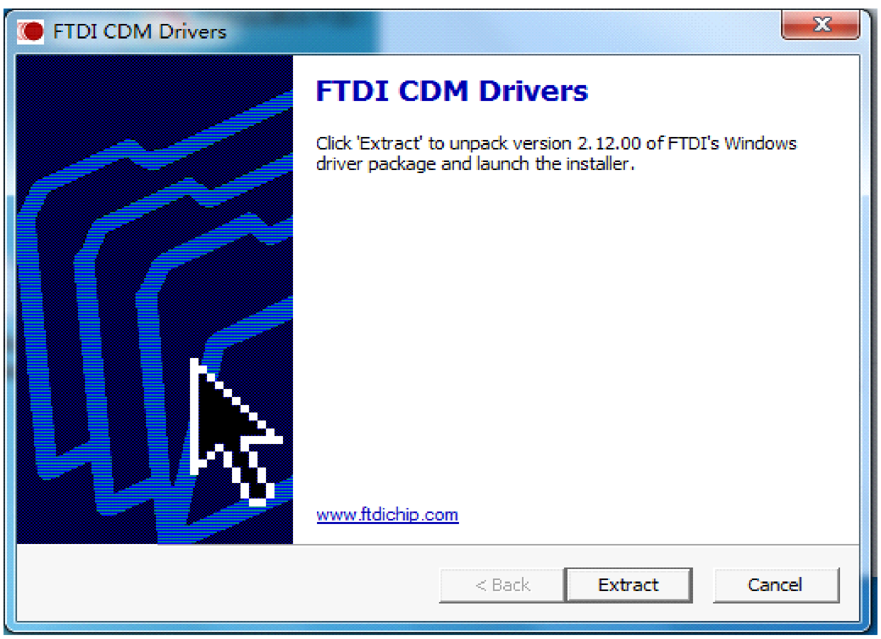
Note: For non-Windows operating systems, download the respective driver from the FTDI website.

http://www.ftdichip.com/Drivers/D2XX.htm

Currently Supported D2XX Drivers:

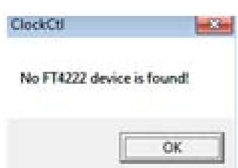
Operating System	Release Date	Processor Architecture: x86 (32-bit)	x64 (64-bit)	PPC	ARM	MIPSII	MIPSIV	SH4	Comments
Windows*	2014-09-29	Available as setup executable Contact support1@ftdichip.com if looking to create cusomised drivers		-	-	-	-	-	2.12.00 WHQL Certified Available as setup executable Release Notes
Windows RT	2014-07-04	1.0.2	-	-	1.0.2	-	-	-	A guide to support the driver (AN_271) is available here
Linux	2012-06-29	1.1.12	1.1.12	-	1.1.12 Suitable for Raspberry Pi	-	-	-	ReadMe
Mac OS X	2012-10-30	1.2.2	1.2.2	1.2.2	-	-	-	-	Requires Mac OS X 10.4 (Tiger) or later ReadMe
Windows CE 4.2-5.2**	2014-22-04	1.0.1.10	-	-	1.0.1.10	1.0.1.6	1.0.1.6	1.0.1.6	
Windows CE 6.0/7.0	2014-22-04	1.0.1.10 CE 6.0 CAT CE 7.0 CAT	-	-	1.0.1.10 CE 6.0 CAT CE 7.0 CAT	1.0.1.6	1.0.1.6	1.0.1.6	For use of the CAT files supplied for ARM and x86 builds refer to AN_319

3. Double click the executable file to install the driver.

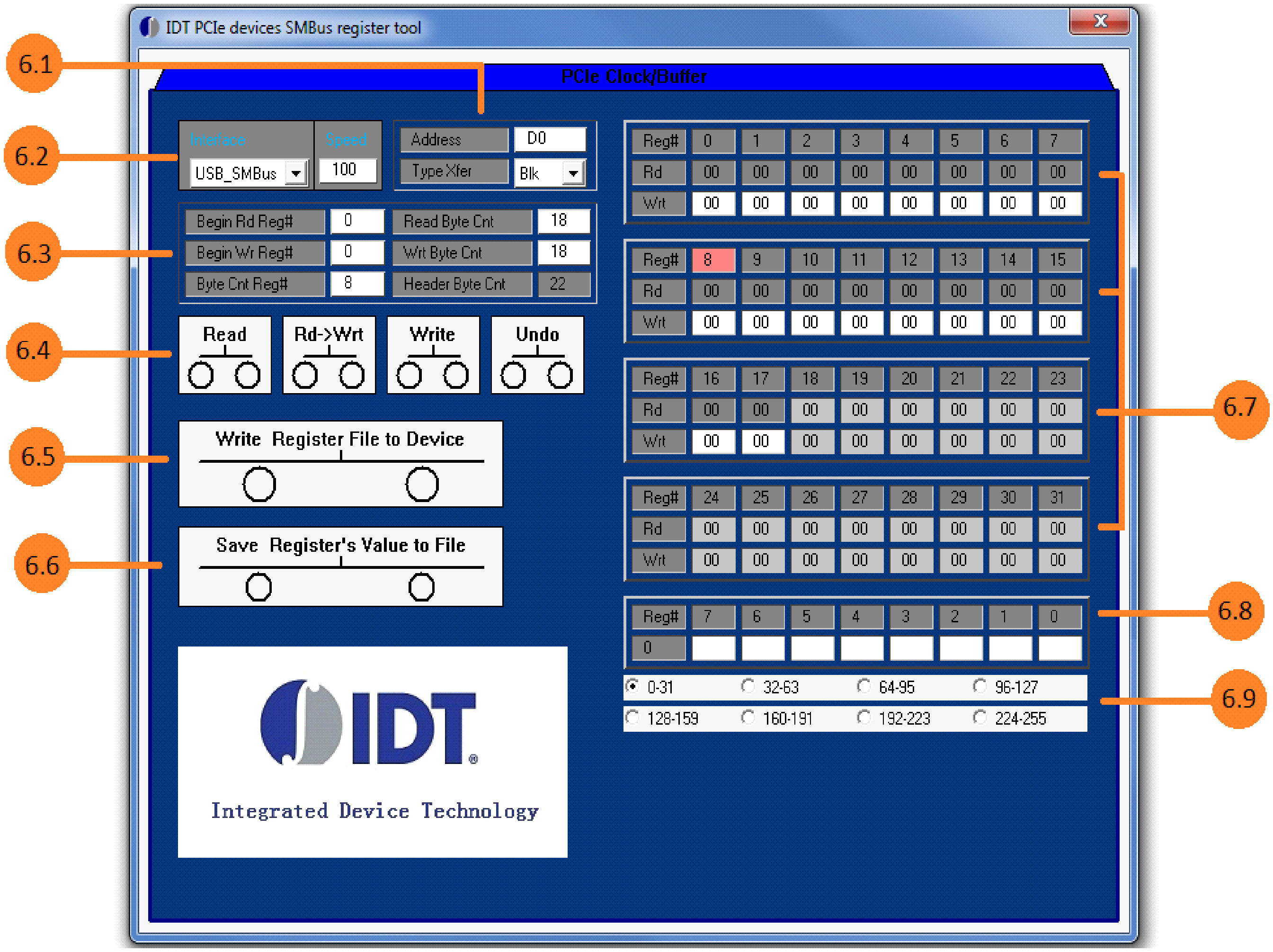


4. Connect the board to the computer using the supplied USB cable. Double click on the Application file ClockCtl.exe to start the PCIe GUI support application.

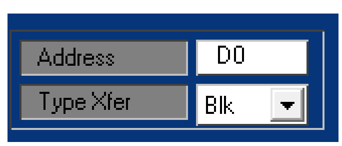
If no board is connected, the following message will appear:



5. PCIe Clock/Buffer GUI main window:



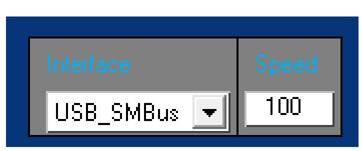
6.1 Slave address



The address is 7-bit slave address combined with 0 in LSB, for example if the slave address is 1101000, D0 should be filled.

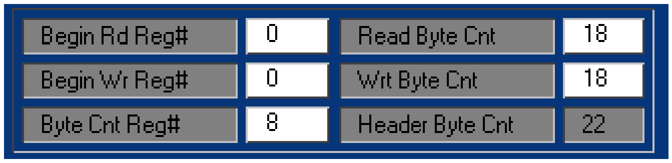
Type Xfer

6.2 SMbus interface



Only USB to SMBus is available, you can change the SMBus speed, but please note that the speed of SMBus is from 10KHz to 100KHz.

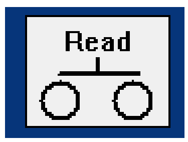
6.3 Begin Reg# and Byte Count



- Begin Rd Reg# is the begin register address of read operation.
- Read Byte Cnt is the byte count of read operation.
- Begin Wr Reg# is the begin register address of write operation.
- Wrt Byte Cnt is the byte count of write operation.

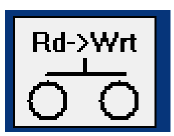
6.4 Register Operations

6.4.1 Read Operation



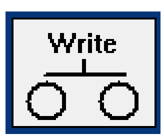
Pressing the read button will initiate a read. If a chipset is used for reading, the byte count is determined by the value in the device byte count register. The byte count cannot be larger than 32 dec. Non-read locations in the read grid will be grayed out.

Rd->Wrt Operation



Pressing the Rd>Wrt button will copy all of the read cells to the write cell contents

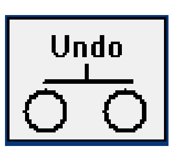
6.4.2 Write Operation



Write button operation. If the chipset is used for writing, the byte count is controlled by the value in the GUI panel byte count register. Registers that will not be written because of the starting location setting and byte count will be grayed out.

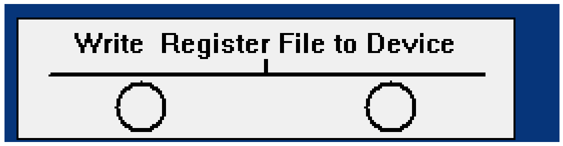
The hex values for data to be written will be in a cell with a white background.

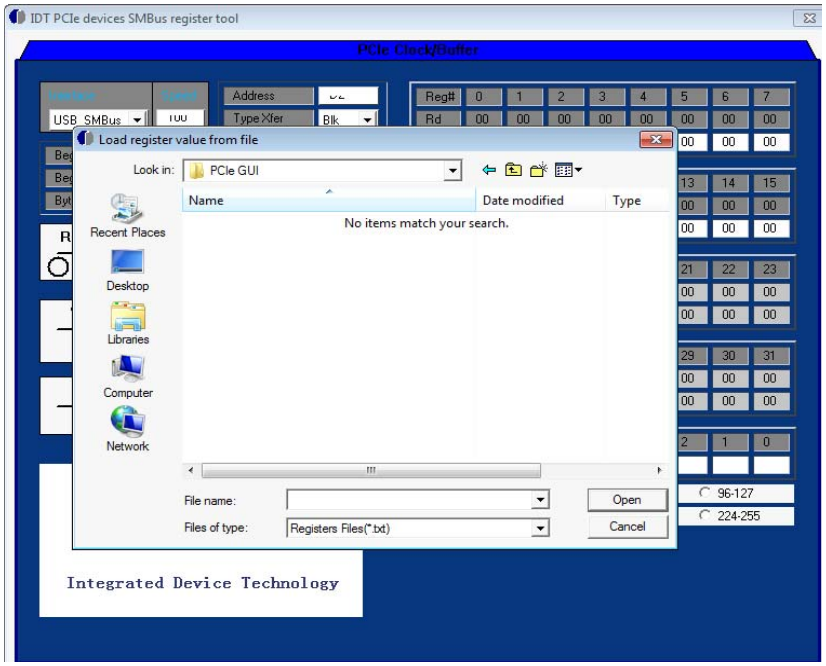
6.4.3 Undo Operation



Reverts the last performed operation.

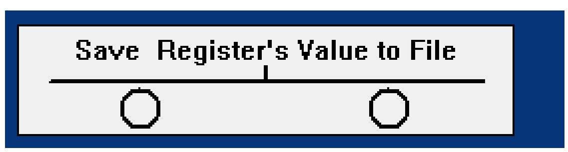
6.5 Write from file

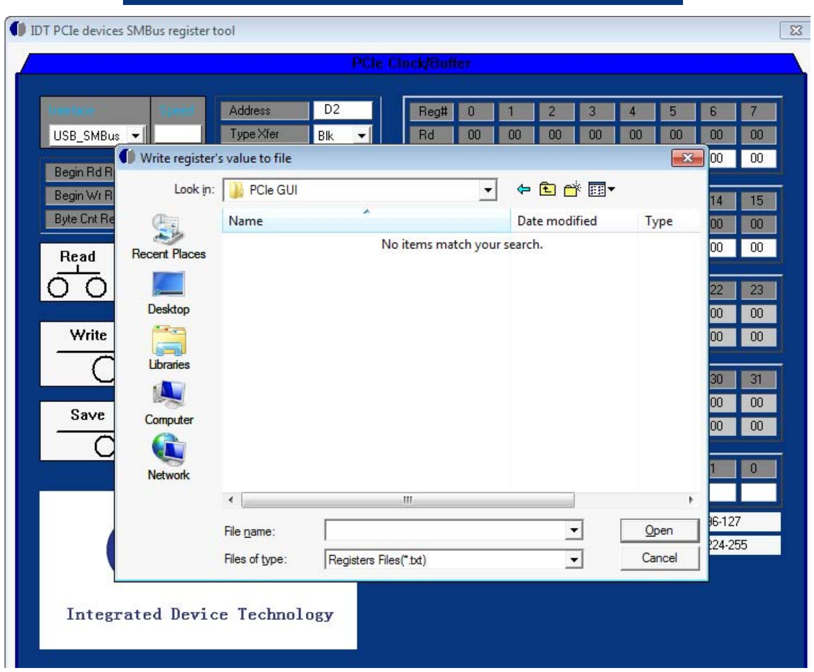




To Write register from file, click "Write Register File to Device" button, it will pop up a window, select the file path and the file name, then click "Open", the GUI will read all registers' value from the file then down load to device.

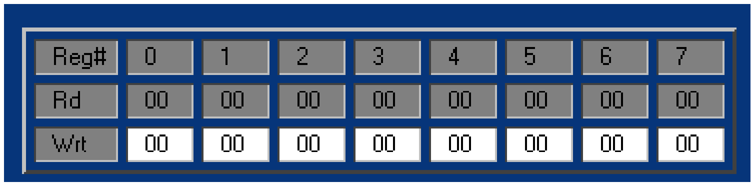
6.6 Save registers to file





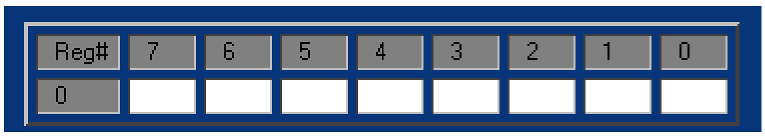
To save registers to file, click "Save Registers Value to File" button, it will pop up a window, select the file path and fill the file name, then click "Save", the GUI will dump all registers' value then save to the file.

6.7 Register Value field



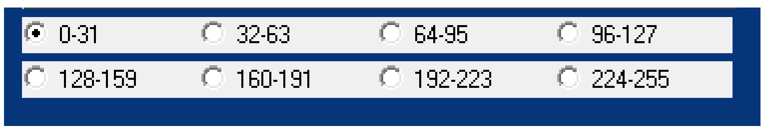
The hexadecimal read information will be grayed background reminding the user that it cannot be altered. Hexadecimal write information will be on a white background.

6.8 Binary display table



Clicking on a Reg# Rd window will display the binary decode of the hex value. This may be used for entering binary data instead of hexadecimal data.

6.9 Byte count range switch



Since there is 32-byte value could be display at the time, if the byte count exceed 32, need to switch the range.

6. Read/Write Operations

Read

Pressing the read button will initiate a read. If a chip set is used for reading, the byte count is determined by the value in the device byte count register. The byte count cannot be larger than 32 dec. Non-read locations in the read grid will be grayed out.

Rd->Wrt

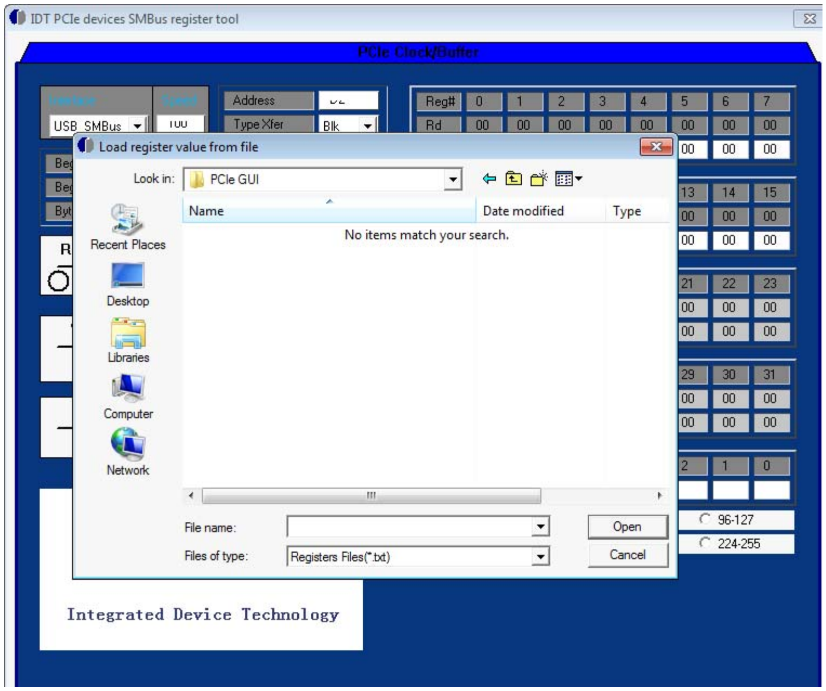
Pressing the Rd>Wrt button will copy all of the read cells to the write cell contents.

Write

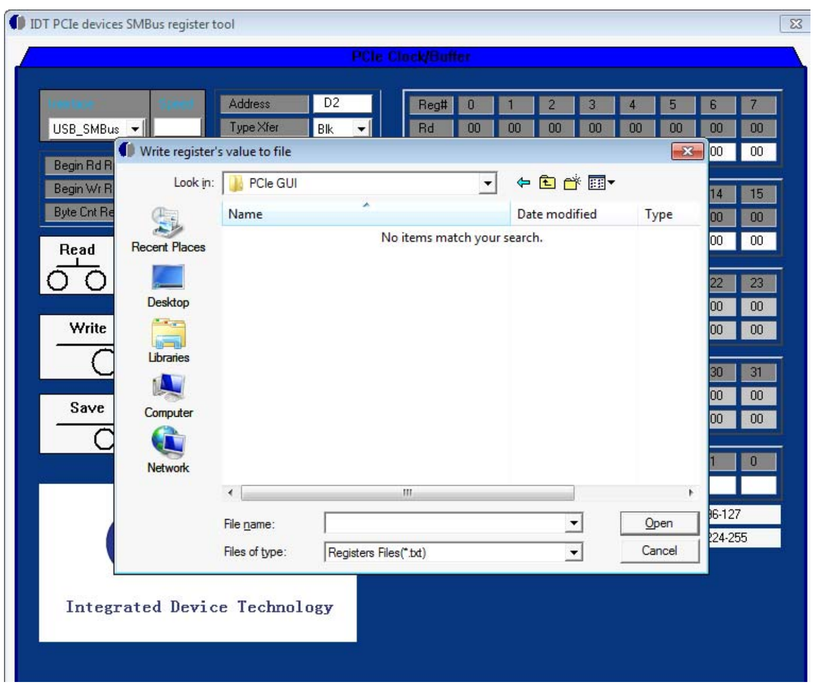
Write button operation. If the chip set is used for writing, the byte count is controlled by the value in the GUI panel byte count register. Registers that will not be written because of the starting location setting and byte count will be grayed out.

The hex values for data to be written will be in a cell with a white background.

7. Read/Write from file



To Write register from file, click "Write Register File to Device" button, it will pop up a window, select the file path and the file name, then click "Open", the GUI will read all registers' value from the file then down load to device.



To save registers to file, click "Save Registers Value to File" button, it will pop up a window, select the file path and fill the file name, then click "Save", the GUI will dump all registers' value then save to the file.

Note: LED LD1 will light up on every SDATA operation.

Board Schematics

Figure 4. 9FGL0841 Schematics

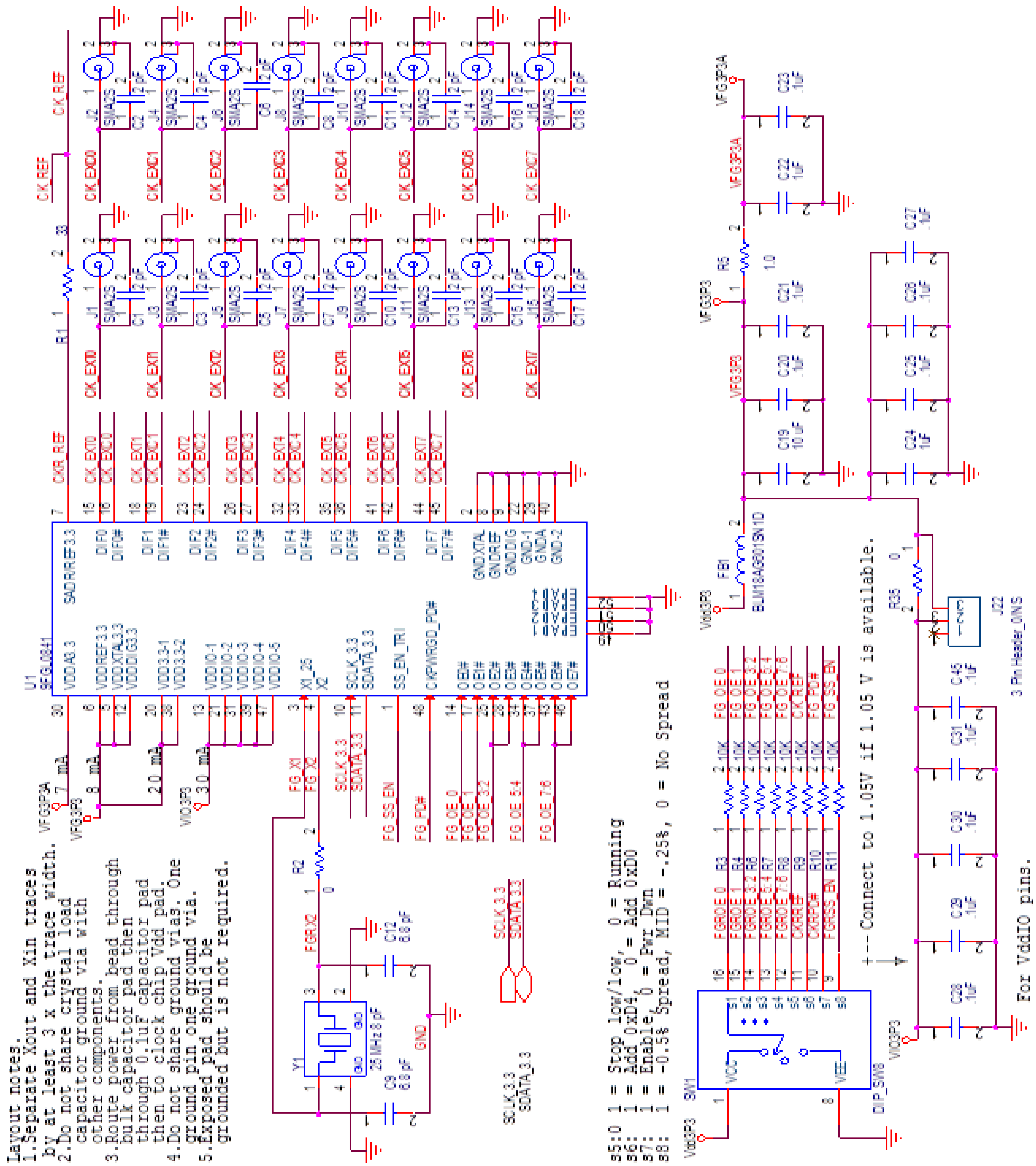
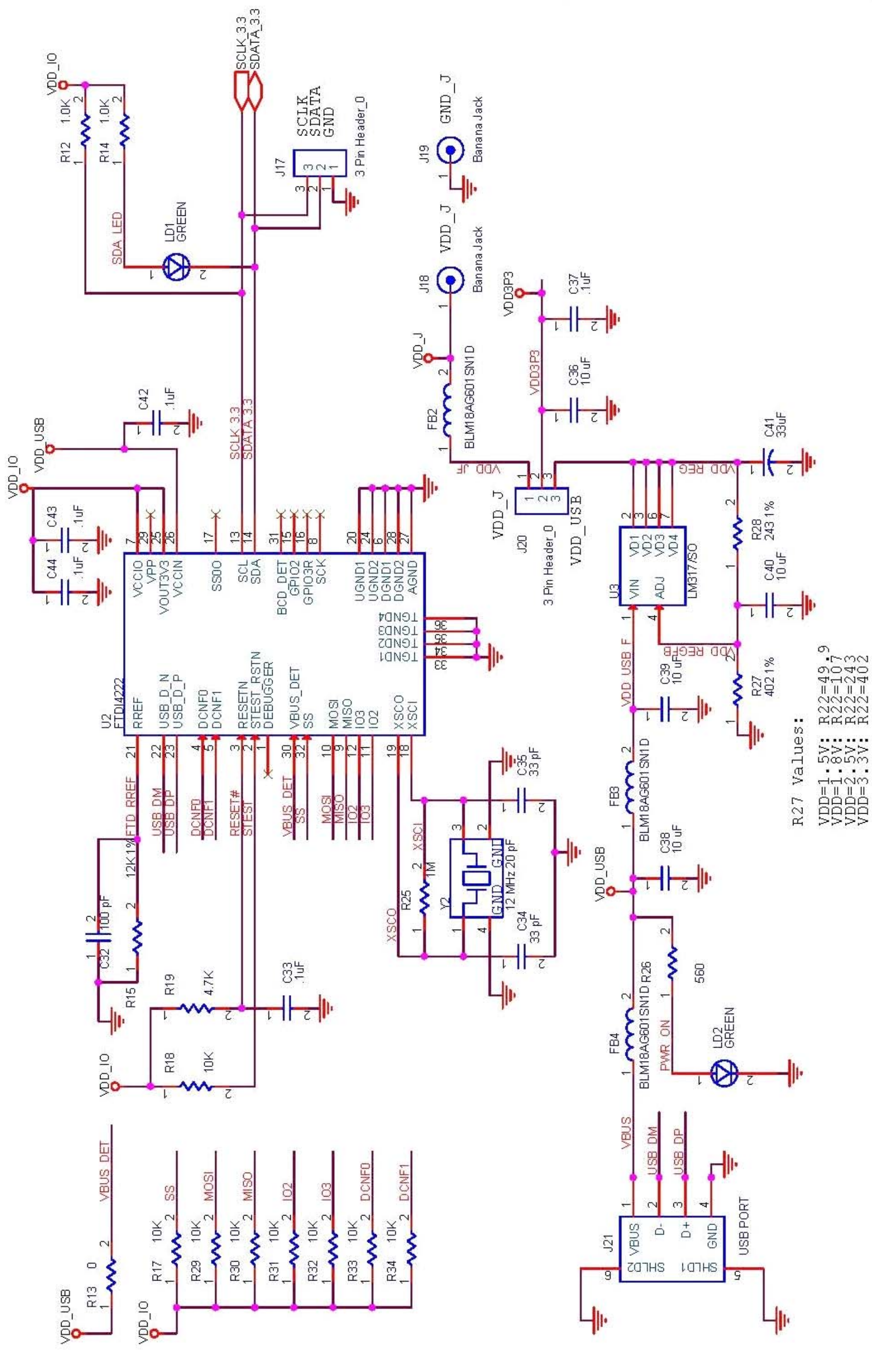


Figure 5. USB Interface and Power



Orderable Part Numbers

The following evaluation board part numbers are available for order.

Table 2: Orderable Part Numbers

Part Number	Description
EVK9FGL0841	9FGL0841 Evaluation Kit
EVK9FGV0841	9FGV0841 Evaluation Kit
EVK9FGU0841	9FGU0841 Evaluation Kit
EVK9FGL0851	9FGL0851 Evaluation Kit