



Introduction

Data acquisition (DAQ) hardware acts as the interface between a computer and external signals, and has many industrial applications. This application note shows how Digital data is transferred from GreenPAK4 SLG46620V to a computer and received by LabVIEW VI. An interface is established between LabVIEW and GreenPAK4 SLG46620V by using a USB to TTL module (FT232RL). After establishing the serial interface, digital data received is indicated by 8 "LED's" in a LabVIEW GUI that are displayed ON/OFF. Featuring 8 Digital input channels, this DAQ using GreenPAK4 SLG46620V can be completed for less than \$10, which is efficient, low cost, and compact.

Brief Introduction of LabVIEW and NI-VISA

LabVIEW

Labview software is used for a wide variety of applications and industries. LabVIEW is a highly productive development environment for creating custom applications that interact with real-world data or signals in different fields.

NI-VISA

The Virtual Instrument Software Architecture (VISA) is a standard for configuring, programming, and troubleshooting various instrumentation systems including Serial interfaces.

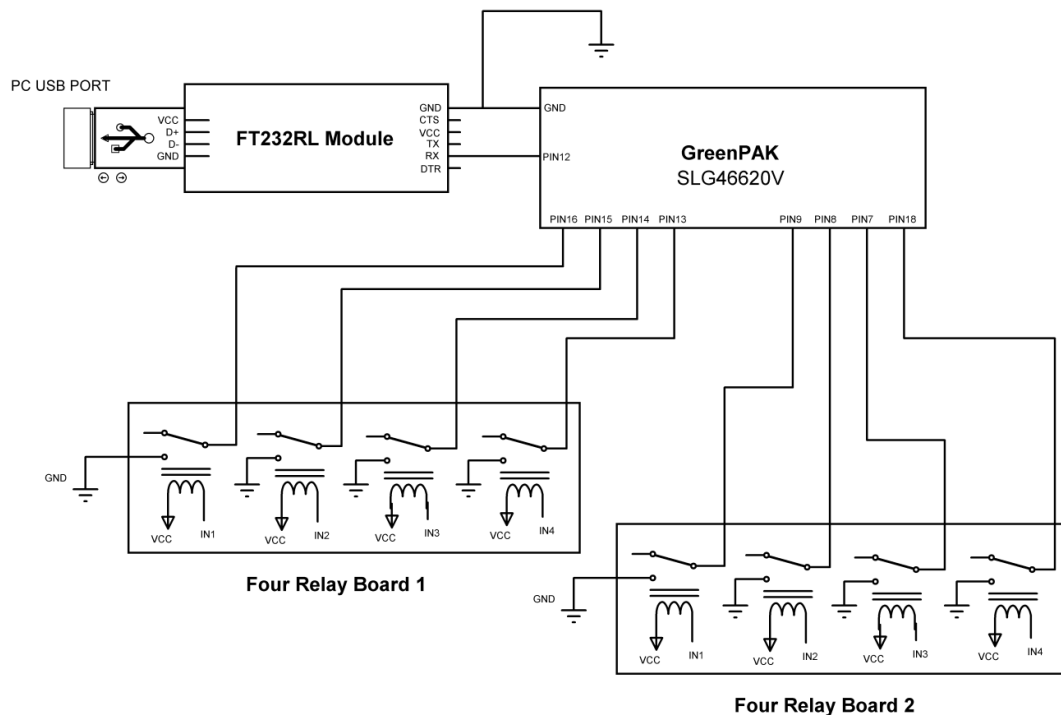


Figure 1. DAQ 8-Ch DI Hardware Schematic

It provides the programming interface between the hardware and LabVIEW. NI-VISA is the National Instruments implementation of the VISA I/O standard.

DAQ Hardware design

The 8-Ch DI DAQ Hardware schematic is shown in Fig. 1.

FT232RL USB to TTL Module

FT232RL USB to Serial Breakout for the FT232RL is a small board with a built in USB to serial UART interface. This little breakout is built around the FT232RL IC from FTDI, with an internal oscillator. Connect the Rx pin of this board to PIN 12 of the GreenPak Development kit. PIN 12 will be programmed as Tx pin in GreenPAK designer and it sends the serial data to LabVIEW VI.

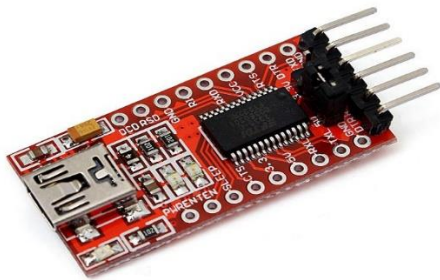


Figure 2. FT232RL USB to TTL Module

Configuring the COM port

When the module is detected as COM port in the device Manager, it can be configured through NI MAX. We are using the default parameters of the COM port. The configuration of COM port can also be changed later in the LabVIEW Programming.

Four Channel 5V Relay Module Board

This module has four Single relays "0SRD-05VDC-SL-C". Each relay is driven by optocoupler 817C. An external 5V power supply is used for VCC and ground pins. SLG46620V digital pins interface with the normally open (NO) contact of the relays. The use of these relays is merely to generate 8 digital signals that the SLG46620V can sample at its inputs and demonstrate functionality.

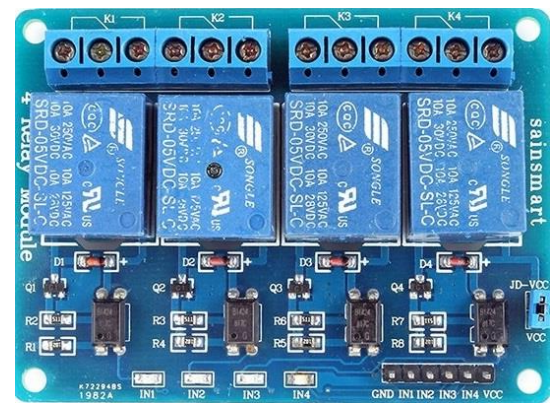


Figure 3. Four Channel 5V Relay Module Board

GreenPAK Design code

The design code consists of four parts

1. Enable Digital Write Operation
2. 4-Bit Mod-10 Counter
3. Reset Mod-10 Counter
4. Concatenating Start bit , 8 digital inputs and Stop Bit

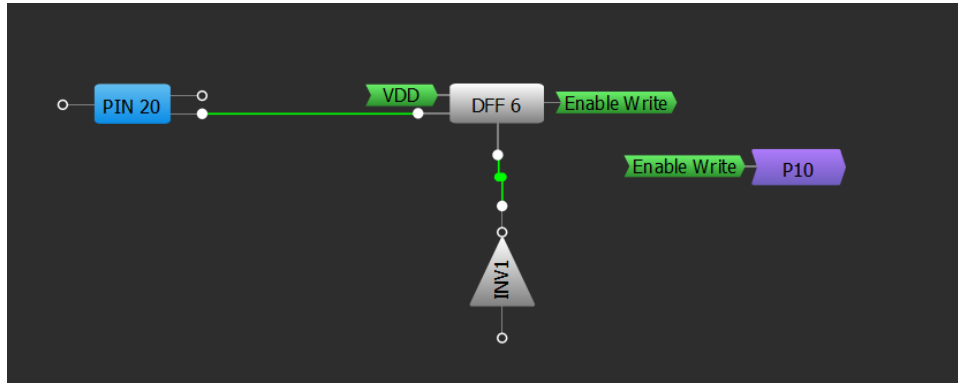


Figure 4. Enable Digital Write Operation

1. Enable Digital Write Operation

This part of the code enables the digital input write operation from GreenPAK SLG46620V to LabVIEW. When PIN 20 is set high, it sets the 'DFF 6' on its rising edge. The reset pin is permanently low through an inverter. The Enable Write is passed to Matrix 0 through P10.

The normal counter in GreenPAK has one output so it cannot be used here. Among the four outputs (Q4, Q3, Q2 and Q1) of the flip-flops, Q4 is MSB and Q1 is LSB.

Also, the circuit counts up to 10 states and then it is reset by the reset block (explained in next section).

2. 4-Bit Mod-10 Counter

This 4-Bit synchronous counter circuit is required to switch control between eight digital inputs, start bit and stop bit.

Workings of a single block of counter

The first block of the counter is enabled by P10 (Enable Write). The XOR gate (2-L0) then sets its output high, which sets the 'DFF 0' output high. CLK is set to have time period of 104µS through OSC and CNT0 block settings.

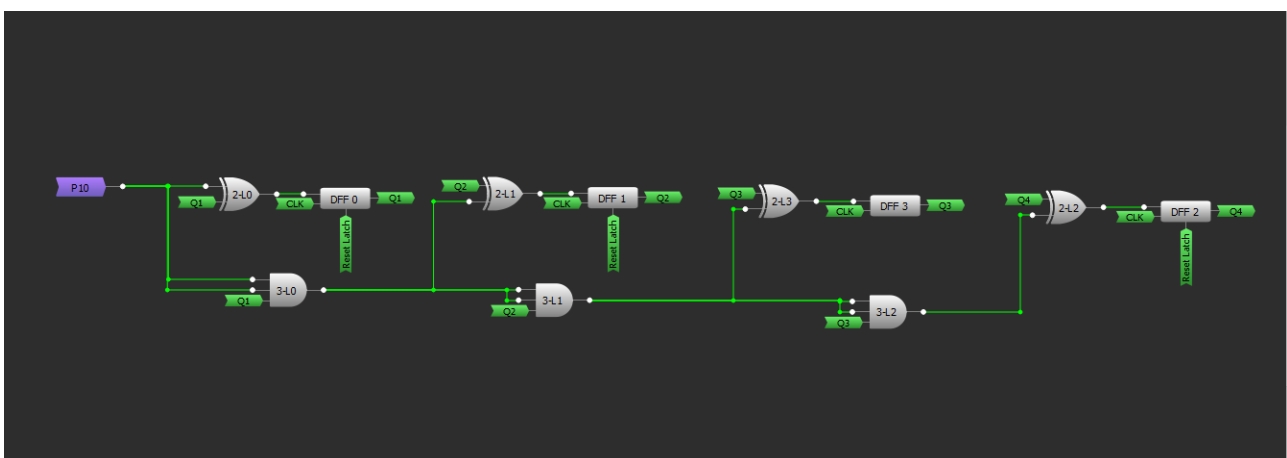


Figure 5. 4-Bit Mod-10 Counter



For every 104μS the clock input checks the DFF 0 block input (coming from XOR gate (2-L0)) and set it high or low. The high Q1 output is then passed through the AND gate (3-L0) for the next block. During the next cycle of CLK, the high Q1 at the input of XOR gate (2-L0) toggles its output to zero and it is passed to the next block.

So the output of first block toggles between 0 and 1 for every clock cycle of the CLK.

As shown in figure 7, the second block of the counter is similar to the first except the enable input is coming from the first block (3-L0 output).

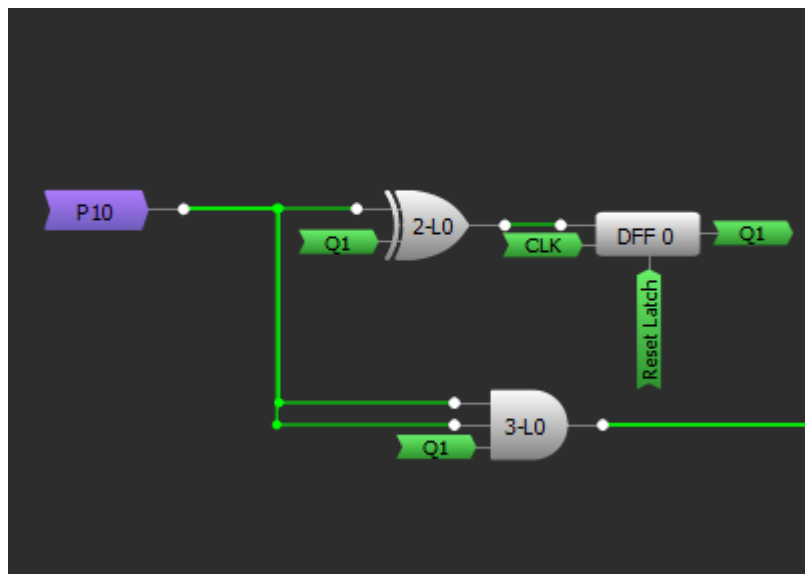


Figure 6. First Block of 4-Bit Counter

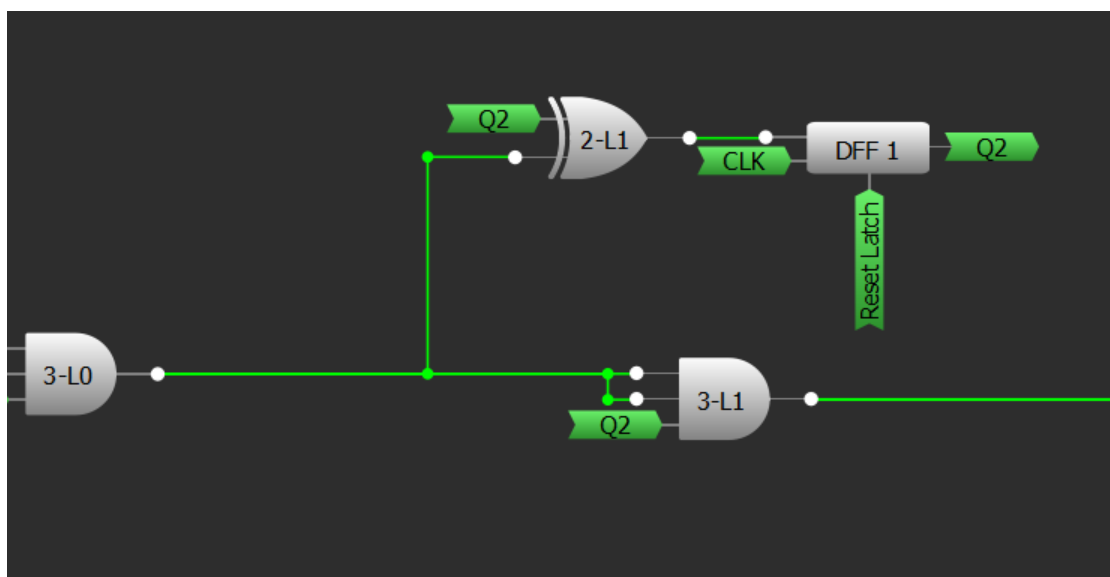


Figure 7. Second Block of 4-Bit Counter



The next two blocks are also similar to the second block.

1. The first block is enabled during every cycle of CLK, hence its output is toggled every cycle.
2. The second, third and fourth blocks are enabled during every 2nd, 3rd and 4th cycles of CLK respectively, hence their output is toggled every 2nd, 3rd and 4th cycles respectively.

3. Reset Mod-10 Counter

During normal operation of the counter (after enabling) it starts counting from 0000, 0001, 0010...1001.

Since we have to process 10 data bits (8 data bits, start and stop bit), therefore after reaching the 10th state (1001), we need to reset the counter to its initial state (0000) in the next cycle. Otherwise it starts counting up to 1111 (16 states) and then overflow.

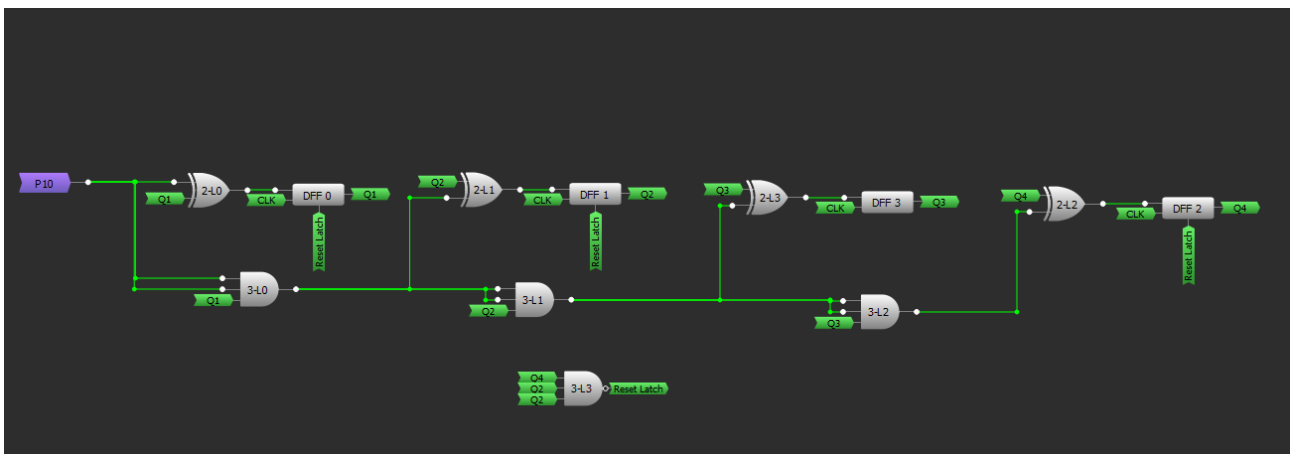


Figure 8. Reset Mod-10 Counter

Sr. No	Q4Q3Q2Q1	Decimal	Data
1	0000	0	Start Bit (0)
2	0001	1	D1
3	0010	2	D2
4	0011	3	D3
5	0100	4	D4
6	0101	5	D5
7	0110	6	D6
8	0111	7	D7
9	1000	8	D8
10	1001	9	Stop Bit (1)

Table 1. Counter outputs and Data bits



The AND gate (3-L3) resets the counter before going to 11th state of the counter where Q4 and Q2 are high.

4. Concatenating 8 digital inputs, Start bit and Stop Bit

In this phase the counter outputs trigger one of the 10 bits (start bit, 8 digital inputs and stop bit) to be routed to PIN 12 one by one. The structure of an example data stream is shown below.

Figure 9 shows that the normal output on PIN 12 should be high. When data starts to be written, a start bit (0) is sent, then 8 data bits and a stop bit (1) at the end of packet.

Basic Building Block of Data stream

In GreenPAK designer we use 3-bit LUT as a basic building block for concatenating the 10 data bits. The truth table of these 3-bit LUT is shown below.

When IN2 is low the output is same as IN1, and when IN2 is high the output is same as IN0. Since we have mostly 3-bit LUT's available, we have to make a combination of them.

Wider input LUT's would have made the design much simpler.

When Enable write is low, a continuous high is present at PIN12. (See LUT 4 truth table in properties).

After Enable write is high and all four outputs of 4-bit counter are low, the IN1 (GND, start bit) of LUT8 is passed through the blocks and appears on PIN12.

As the CLK starts incrementing the 4 output pins (Q1, Q2, Q3, Q4) the D1, D2...D8 starts appearing on PIN 12, one by one, for one cycle each (104 μs).

IN 2	IN 1	IN 0	Output
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

Table 2. 3-bit LUT Truth Table

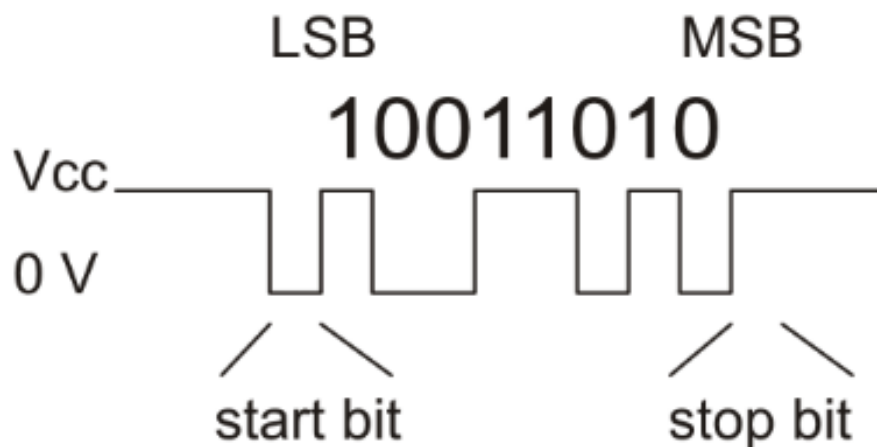


Figure 9. Example Serial Data stream

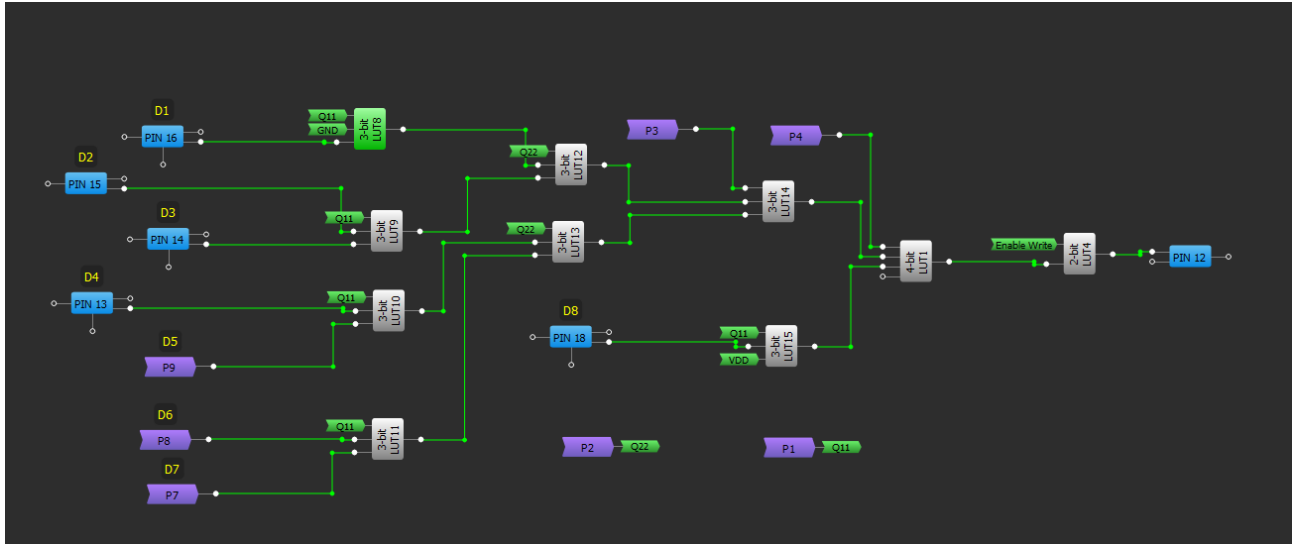



Figure 10. Concatenation of Data Stream

After D8, a stop bit (VDD) will appear at the output. This completes the serial data stream of 8 digital inputs. After 10 cycles of the CLK, all 10 bits (start bit, 8 Digital Inputs and stop bits) are output to PIN12. Counter is reset (see "3. Reset Mod-10 Counter") and it continues to loop output data at PIN12 for every 10 cycles of CLK.

This data reaches LabVIEW after passing through the Rx pin of the FT232RL module. The data which is sent at 104µS per bit is received in LabVIEW at a baudrate of 9600 bps (approx. 104µS per bit).

Running the LabVIEW GUI

Now insert the FT232RL Module and go to front panel. Click drop down menu of select COM port, press refresh and select the COM port. It's COM6 in this example but may be some other number in your case. Also at that point the Tx pin of the module shall be connected with the PIN 12 of the GreenPAK development kit and Digital Write operation must be enabled from GreenPAK chip (PIN 20 must be high).

After selecting the COM port, press Run Continuously . It is on the top right side of the panel. It must be noted that the COM port must be selected before pressing Run Continuously, otherwise the program will misbehave and then it must be closed from Task Manager.

Now turn ON/OFF different Digital pins from GreenPAK emulator (button option) and watch the corresponding LED ON/OFF in LabVIEW.

Example

When heavy industrial loads are operated, it's usually by relays which may have unused contacts available. The extra contacts can be used as a source for the indication of load status (ON/OFF). The GreenPAK ground can be connected to one contact, while the other contact goes to a GreenPAK input, which has an internal pullup programmed in. When the contact is open (load OFF) the digital input reads high, and when contact is closed (load ON) the digital input reads low. The digital pin is then read by LabVIEW via GreenPAK chip and an LED is turned ON/OFF in VI.

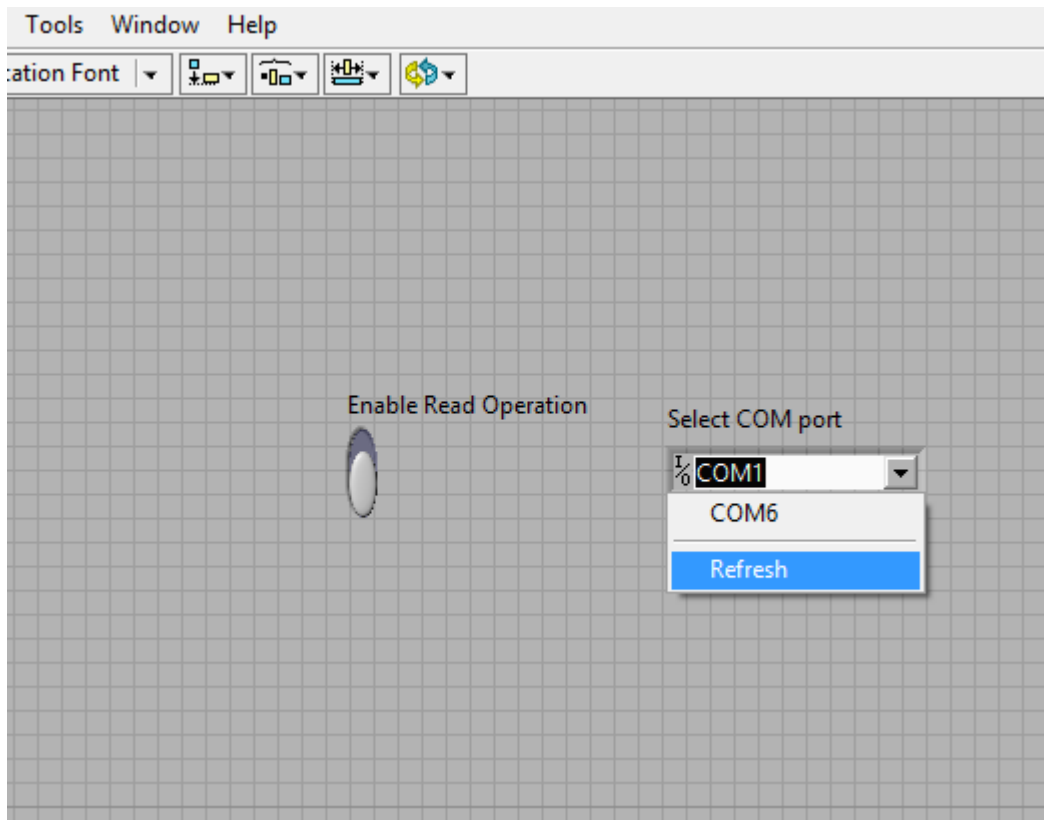


Figure11. Select COM port in LabVIEW

The scenario is simulated (in attached video) with eight Relays from the board described earlier. The corresponding LED state changes with relay operation. Proper grounding of the high voltage side must be ensured before attaching digital inputs of GreenPAK chip with the possibly available contacts. This is a necessary design consideration for general safety and all the hardware.

Limitations

This design offers a low cost DAQ solution limited to DO channels, and the DO update rate is limited to the COM port baud rate of 9600 bps (approx. 10kHz). Care should be exercised when connecting GreenPAK inputs to relay contacts.

A high voltage induced due to improper grounding or inductive spikes on the contact side of the relay may exceed the ratings of the GreenPAK inputs, causing a safety hazard.

Conclusion

This application note demonstrated the usefulness of dedicated DI channel only, for industrial controls. While it cannot update faster than 10kHz, Windows applications usually require 200 milliseconds to update anyway. So the COM port baudrate of 9600 bps is comparatively adequate.

The cost advantage of this design makes it an attractive choice for dedicated controls, allowing many more installations.



Also, the design using the SLG46620V can be made quite compact, which allows placement where generic DAQ simply would not fit. Additionally, this application example along with "8 Channel Digital Output Data Acquisition Card using Labview" can be combined to make a complete Industrial DIO solution.

References

1. Advantages of Using LabVIEW

<http://www.ni.com/white-paper/8534/en/#toc7>

2. National Instruments VISA

<https://www.ni.com/visa/>

Appendix

LabVIEW Programming Guide (for the application note)

After creating a blank VI, go to NI LabVIEW front panel and right click, the Control window opens. Select Modern >> I/O >> VISA Resource.

For adding Enable read operation button, again right click in LabVIEW Front panel, Select Modern >> Boolean >> Vertical Slide switch.

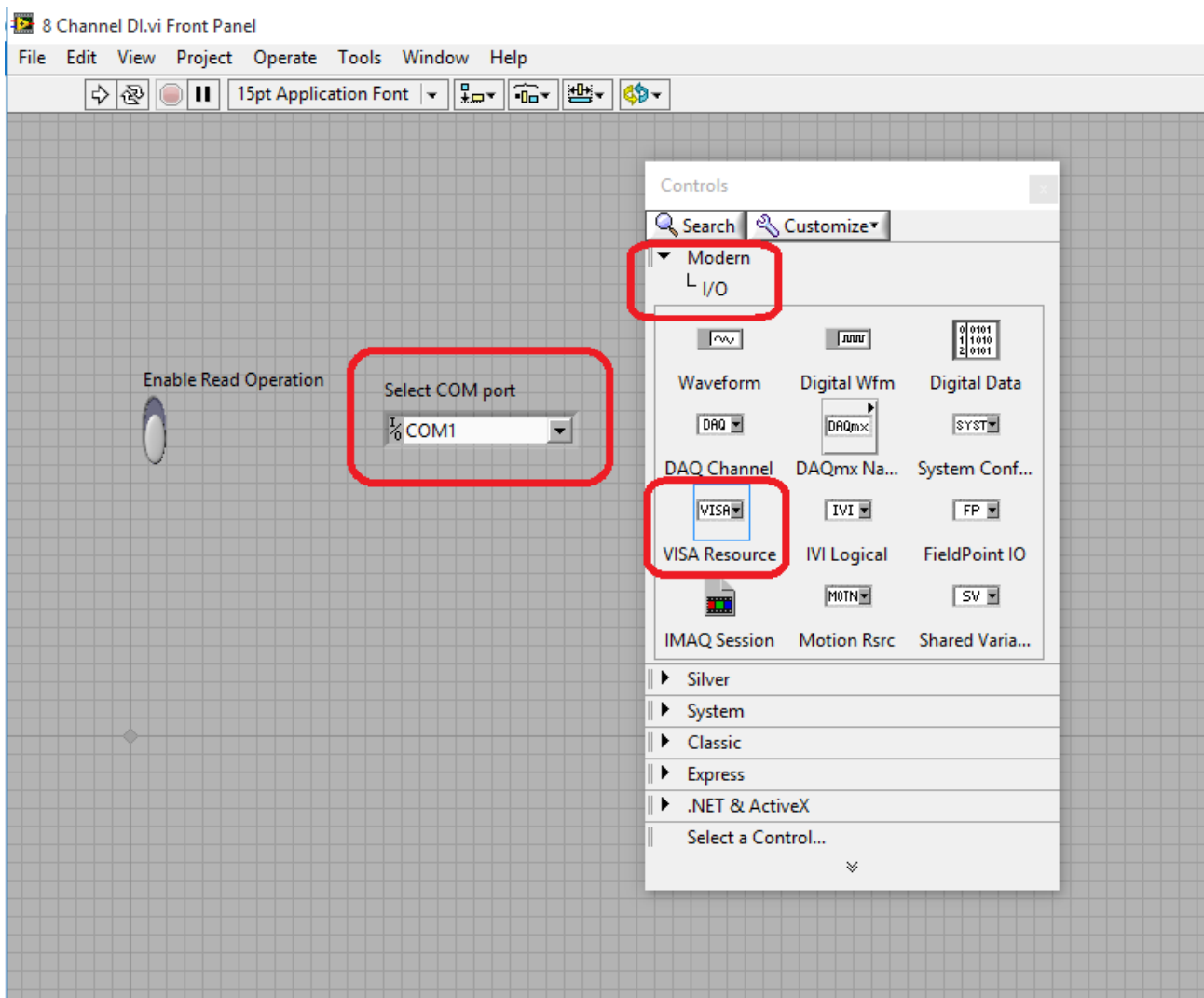


Figure 12. Adding COM port in LabVIEW Front Panel

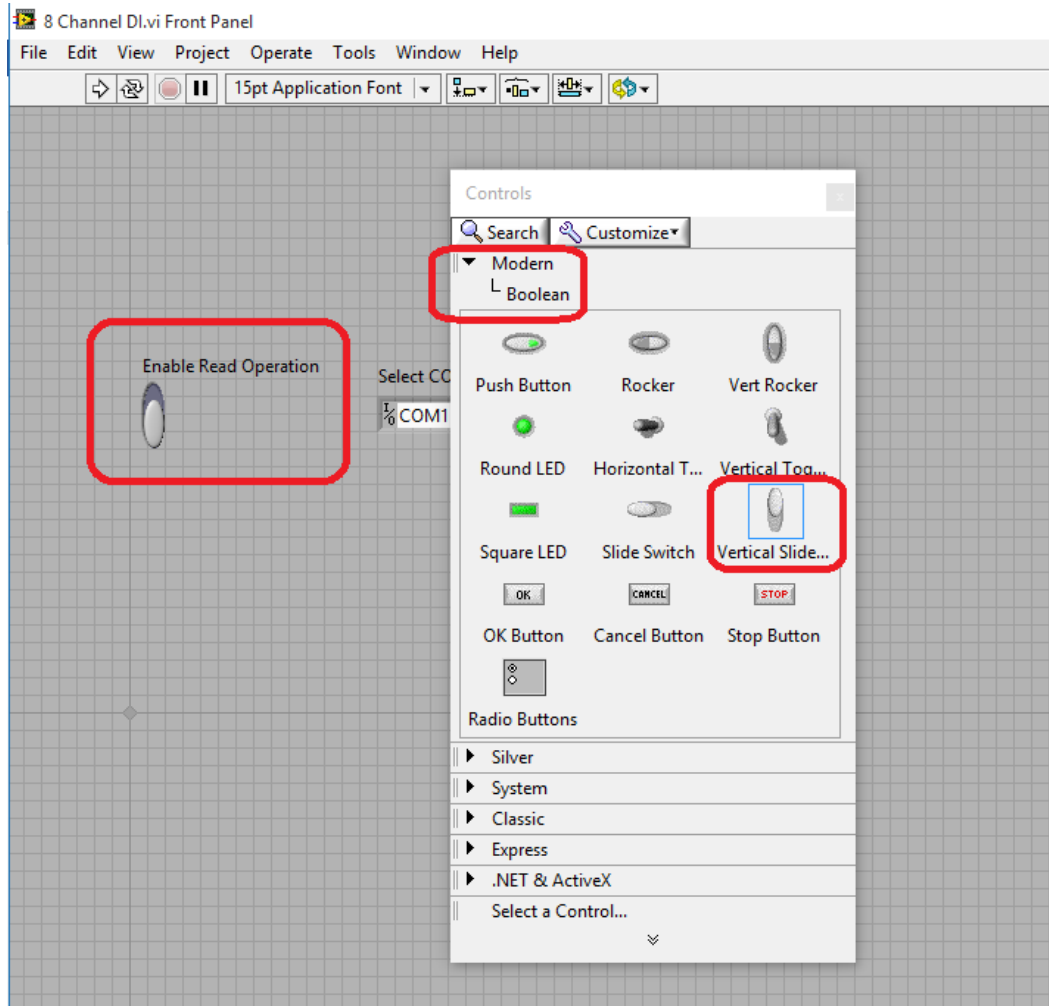


Figure 13. Adding Vertical Slide switch in LabVIEW Front Panel

Next step in the LabVIEW programming is to join these components to receive data from GreenPAK4 SLG46620V. For this go to Block Diagram, (Press CTRL + E when in Front Panel, CTRL + E toggles the screen between Front Panel and Block Diagram).

Add numeric constants, Case structure, VISA Serial, VISA read, VISA Close, Unsigned Byte Array, Index Array, Number to Boolean Array and Array to Cluster functions in the Block Diagram. These components can be searched in the search box in the right top corner and then added one by one.

By hovering the cursor at the output of Array to Cluster and right clicking >> create >> indicator. An eight row LED display appears in the front panel (it can be resized to more or less number of LED's).

The COM port is configured as shown in the Figure 14. The wiring can be done by simply hovering the cursor over the connection and then left clicking and dragging towards the target connection.

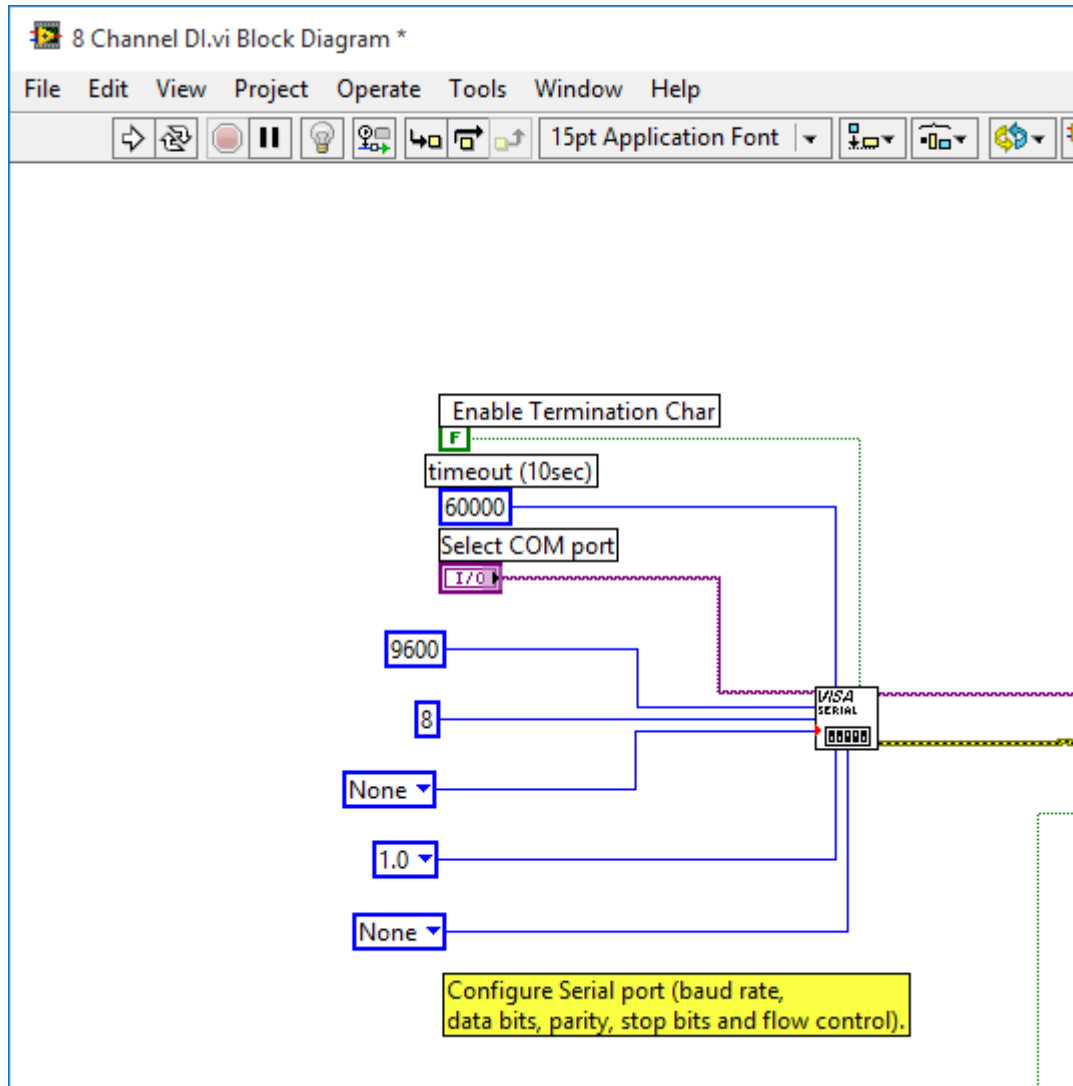


Figure 14. COM port configuration in LabVIEW

Here we can change the default COM port settings set in the NI MAX.

The Enable Read Operation button is used to select the CASE structure. Next step is to configure the TRUE window of CASE structure. When the connection to the COM port is established, the control is transferred to this window and the data is read from the COM port.

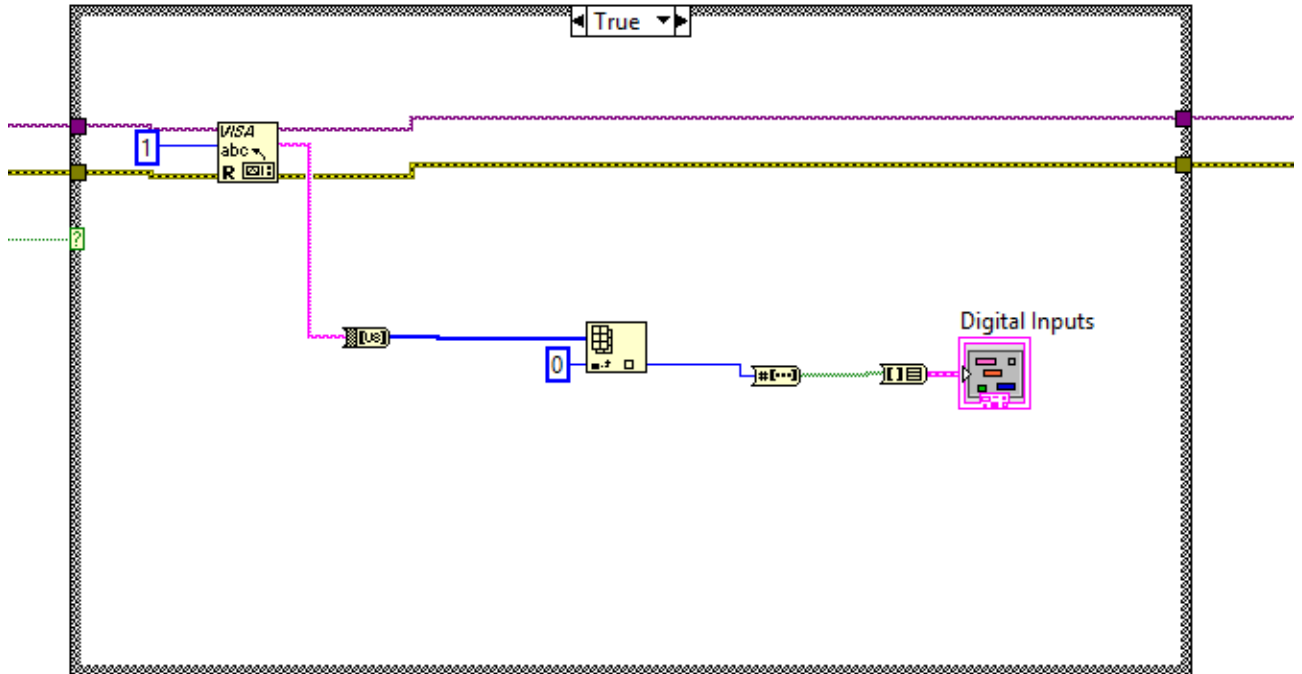


Figure 15. COM port Read configuration in LabVIEW

The data read from the COM port is a string and it is converted into unsigned byte array and then into a Boolean array. The result is then displayed as a cluster of eight LED's. After reading data from COM port, the COM port is closed so that it can be used by other applications without quitting LabVIEW.



About the Author

Name: Bilal Ahmed

Background: Bilal Ahmed received an MS in Electronics Engineering from NED University, Karachi. He has more than 10 years of experience in PLC, SCADA, Data acquisition, Industrial automation, Robotics and Embedded systems. Currently he works as Head of Embedded systems and Robotics in HF Electronics.

Contact: [**appnotes@silego.com**](mailto:appnotes@silego.com)



Document History

Document Title: 8 Channel Digital Input Data Acquisition Card using LABVIEW

Document Number: AN-1131

Revision	Orig. of Change	Submission Date	Description of Change
A	Bilal Ahmed	10/26/2016	New application note

Worldwide Sales and Design Support

Silego Technology maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the sales person closest to you, visit us at **Sales Representatives and Distributors**.

About Silego Technology

Silego Technology, Inc. is a fabless semiconductor company headquartered in Santa Clara, California, with operations in Taiwan, and additional design/technology centers in China, Korea and Ukraine.

