



Certified Experts in Automation Engineering to Design, Control, Test & Adapt

Touch Panel Interface for Wireless Lighting Control Using a Multiple MODBUS Master

Authors:

David W. Lenkner, Ph.D., Automation Systems Engineer, Data Science Automation, Inc.

Gregory C. Cala, Ph.D., Partner & Chief Technology Officer, Data Science Automation, Inc.

NI Product(s) Used:

LabVIEW 2011

LabVIEW PDA Module

MODBUS Library for LabVIEW

Category:

Machine Control

Oil, Gas, and Energy

The Challenge

Create custom software to access set points for lighting control relays from touch-panel Windows CE devices, integrating these devices into an existing MODBUS-based lighting and HVAC (Heating, Ventilation and Air Conditioning) control system.

The Solution

We leveraged NI's comprehensive and user-friendly MODBUS Library for LabVIEW to create a modified protocol which allows multiple devices reading from a single slave process, while still maintaining compatibility with existing MODBUS slave processes.

Introduction

As a premier National Instruments Alliance Partner, Data Science Automation (DSA) specializes in automating and educating the world leading companies. We were sought out for this project due to our deep talent pool of Certified LabVIEW Architects and Certified Professional Instructors, and our extensive experience with National Instruments products.

Reasons Behind the Design

The customer had previously invested in custom I/O controller boards and an accompanying programming environment as part of a building HVAC control system, and had kept the software on these boards fairly general in order to try to facilitate reuse and adaptation later for an integrated HVAC and lighting control system. However they had failed to anticipate future need for "auxiliary" touch panel devices to directly access the MODBUS slaves. This functionality is not supported by standard serial MODBUS communication, which is the protocol that had been chosen previously for the lighting control system.

In order to overcome these limitations, DSA considered architectures that would modify the master MODBUS programs without requiring adjustments to the slave systems. This is because the customer's legacy MODBUS slave processes had been implemented via specialized non-NI products and were difficult to redesign, having their own non-standard programming environment which lacked the modularity, extensive documentation, and ready-to-use libraries of LabVIEW.

On the other hand, the MODBUS master console running on the PC was written in NI LabVIEW, and LabVIEW also supports executable builds for the ARM7 Windows CE 5 touch-screen devices to be added into the system. This sector of the software used NI's MODBUS library in LabVIEW, and was

much easier to adapt to the customer's particular needs while still being versatile enough to interface seamlessly with the older controller hardware/software.

Multiple-Master MODBUS (MMM) Read Solution

Our solution utilized the fact that the single Windows 7 master program already cycled through each slave to collect and log lighting relay conditions over time. Therefore, the data that was desired by the additional master touch-panel devices was already present on the serial line at various times, and the extra read points did not need to actually send MODBUS commands, but need only wait for the appropriate MODBUS response. These processes may be referred to as "passive" MODBUS masters.

However, there was some complication regarding how the passive master processes should determine exactly which MODBUS message was the correct one to process. In the traditional MODBUS messaging system, the passive read devices might mistakenly accept a MODBUS read response from the wrong register address, only to have the correct read response arrive later.

Therefore, to differentiate between read responses from a given slave, we designed a system where every block of memory to be read from the slaves would begin with a "key" address. This holding register on the slave would be set to a known value not having any physical significance to the relay values, but would be unique to only that block of memory addresses, and which would be known beforehand (via a static configuration file) by the master processes. Upon receiving a MODBUS read response, the master process would check that the "key" address contained the expected value, and if it did not, it would discard that response and await a further message. This resulted in a MODBUS read function that took the usual parameters (slave identity, register address, etc.) as well as a "key value" or expected value for the first register in the array, as shown in Figure 1.

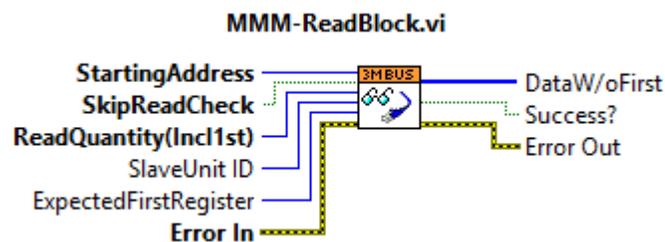


Figure 1 - Custom VI for "passive" read of data from MODBUS slave without actively sending read request. The 'Expected First Register' argument allows the module to check the "key" value to filter all but the desired response, as described above.

Multiple-Master Write Capabilities

The customer indicated that the touch-panel modules (Figure 2) would not necessarily require MODBUS writing ability, since light switches would be mounted nearby for adjusting throughout the building, and the main PC process would provide a centralized access location. However the customer did express that it would be a valuable added benefit to have limited write ability to assist in diagnostics and discovery during installation and configuration of the MODBUS devices. Therefore, some (limited) write functionality was built in to the multiple-master MODBUS module.

In order to allow some multiple-master write capabilities for testing, we allowed the customer to activate a random-timed write mode where two or more masters would attempt to speak to a slave at the same time. This mode allows the possibility of masters "stepping on" each other's commands, but in a situation where only a few masters are speaking at a time (say, the main PC control program as well as an additional master touch panel being currently installed) then this is not likely to occur. If it did occur, we monitored CRC (Cyclic Redundancy Checking) error checking in the MODBUS responses to reject any invalid communications that failed in this way. CRC is conveniently included in the NI LabVIEW MODBUS library and required no additional programming on our part.

Benefits of Partnership

The Alliance partnership between Data Science Automation and National Instruments has allowed DSA to tackle many complex engineering challenges while maximizing our client's return on their investment. In this case, the customer's previous investments might have been rendered obsolete were it not for the flexibility of NI LabVIEW and its interface to MODBUS communication. The software using LabVIEW's Windows CE touch panel module allowed for a backwards-compatible solution that satisfied our customer's needs while minimizing development and integration time requirements.

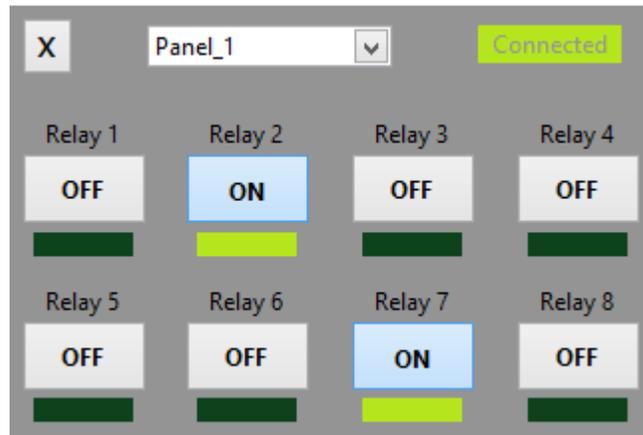


Figure 2 – A touch panel screen for lighting control application. The dark and light green LEDs indicate the read-out relay status while the buttons allow for overwrite of the set points for diagnostic purposes.

Contact Information

David W. Lenkner, Ph.D., dwl@DSAutomation.com
Data Science Automation Inc., 375 Valley Brook Road, Suite 106, McMurray PA 15317
(724)942-6330
www.DSAutomation.com