
AC 2011-1607: CONVEYOR CONTROL SYSTEM PROJECT

David R. Loker, Pennsylvania State University, Erie

David R. Loker received the M.S.E.E. degree from Syracuse University in 1986. In 1984, he joined General Electric (GE) Company, AESD, as a design engineer. In 1988, he joined the faculty at Penn State Erie, The Behrend College. In 2007, he became the Chair of the Electrical and Computer Engineering Technology Program. His research interests include wireless sensor networks, data acquisition systems, and communications systems.

Robert Weissbach, Pennsylvania State University, Erie

Robert Weissbach is currently an associate professor of engineering and incoming director of the Applied Energy Research Center at Penn State Erie, The Behrend College. Prior to completing his doctoral studies, he was employed by General Dynamics Electric Boat Division where he worked on the design and construction of submarine turbine generator sets. From October 2007 through June 2008, he was a visiting researcher at Aalborg University in Aalborg, Denmark. Dr. Weissbach is a Senior Member of IEEE and is a registered engineer in Pennsylvania. His research interests are in renewable energy, energy storage, power electronics and power systems.

Adam Henry, Penn State

Adam Henry is currently a senior at Penn State Erie, The Behrend College majoring in the Electrical and Computer Engineering Technology program. His interests are computers and related technology. He also enjoys designing and building electrical devices to use in his daily life.

Conveyor Control System Project

Abstract

Part of the requirements in a junior-level measurements & instrumentation course (for an Electrical and Computer Engineering Technology program) includes an end-of-semester design project. One possible project is the design and implementation of a conveyor control system. For this project, the conveyor simulator is pre-fabricated and equipped with a 24-volt DC motor mounted directly to a plastic spindle for driving the conveyor belt, a freely rotating plastic spindle for the opposite side of the conveyor belt, and two Plexiglas sides. Three IR reflective sensors (one in the middle and one at each end) are used to detect the position of an object on the conveyor. The objective of the project is for students to design the software (using LabVIEW) and hardware interfacing electronics for the conveyor control system such that it mimics the operation of a conveyor with beginning, stamping, and ending stations. Students are required to use a National Instruments data acquisition system with analog I/O and digital I/O capability. This paper provides a detailed listing of the engineering requirements for the system, the functional test procedure for verifying proper operation of the system, and results. In addition, since this conveyor control system is also a lab project for a sophomore-level PLC course, results are provided for the PLC-based control system along with an assessment based on the comparison of the advantages and disadvantages for PC vs. PLC control. Recommendations are included to help ensure student success on the project. The PC-based control system project has been found to be an effective end-of-semester project for two reasons. First, it integrates both hardware and software design while utilizing information covered from prerequisite courses. Second, due to the slow time response characteristics of this system, PC-based control is suitable for this application.

Introduction to the Measurements and Instrumentation Course

This is a required first semester junior-level course for Electrical and Computer Engineering Technology students. Details about the course are available¹. A primary objective of the course includes using LabVIEW and data acquisition (DAQ) cards for the design of both measurement systems and control systems²⁻³. Rather than requiring students to take a final exam, this course requires an end-of-semester design project. One possible project is the design of an automated test and measurement system¹. An alternative option is for students to design a conveyor control system, and students can choose the project that is most interesting. The purpose of this paper is to describe the details about this conveyor control system project.

Conveyor Control System Project

Students are provided with a prebuilt conveyor simulator and a PC-based DAQ system containing analog I/O and digital I/O capability. The objective is to design a control system for the conveyor in order to meet a set of practical engineering requirements.

This project is appropriate for a measurements and instrumentation course since it requires both hardware and software design for interfacing the conveyor to a PC-based DAQ system for a measurement and control application. Infrared proximity sensors are used for detecting the position

of an object on the conveyor. The time response characteristics for this system are sufficiently slow such that control theory for dynamic control applications is not needed⁴.

Conveyor Simulator

The conveyor simulator uses a 24V dc motor that draws approximately 300mA, Plexiglas side walls, sandpaper for the track, and three OR500-ND infrared proximity sensors⁵. A pre-made part gets placed in front of the first infrared sensor which will start the dc motor. The part will continue down the conveyor at an approximate speed of 1 inch/second until a second infrared sensor is reached and the conveyor will stop. The part will be stamped by the user, and then the part will continue down the conveyor until the third and final infrared sensor. Then the conveyor will stop operation and wait for another part to be placed in front of the first sensor. A drawing of the conveyor is shown below in Figure 1.

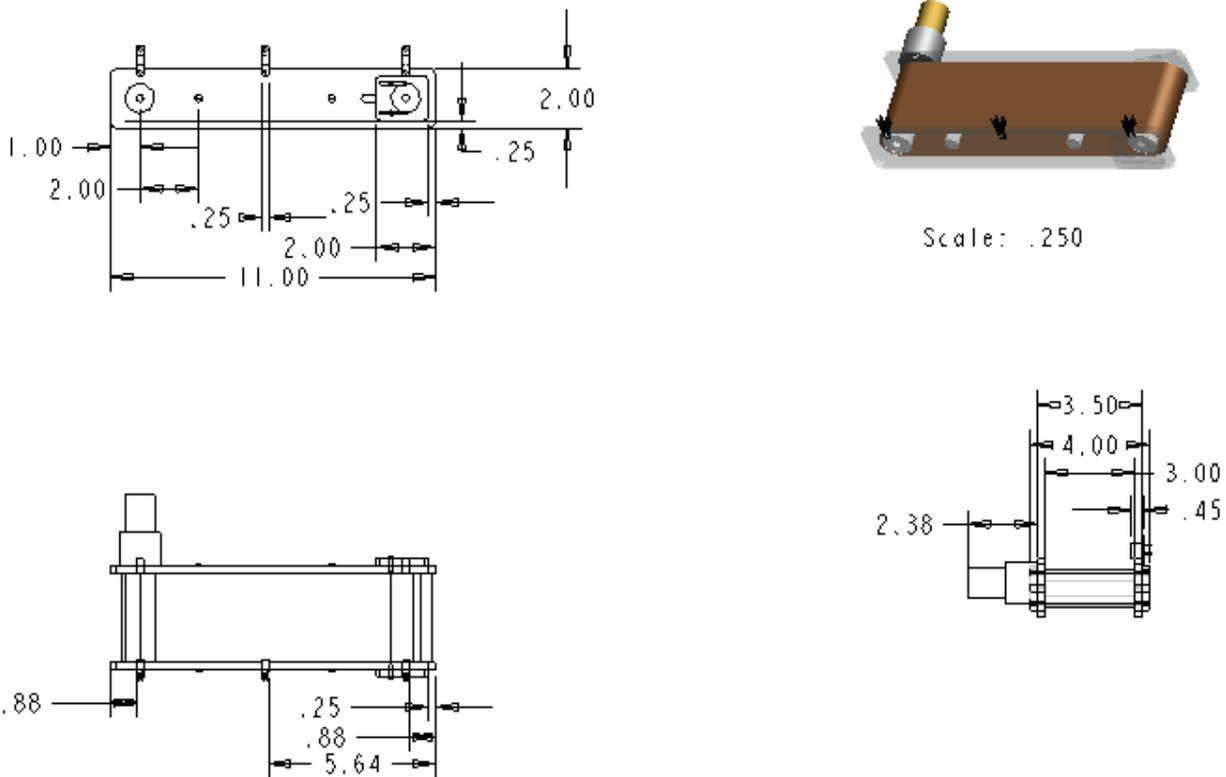


Figure 1. Conveyor Simulator.

Engineering Requirements for project

Special features:

- LabVIEW software development environment
- Pre-built conveyor simulator used for the control system
- Conveyor simulator interfaced to the NI USB-6008 DAQ card⁶

User Interface Requirements:

- A Boolean control on the front panel to indicate that the conveyor is ready to be started
- A display on the front panel to indicate the state of the conveyor
- A display on the front panel to indicate if a part is on the conveyor
- A button (or toggle switch) on the front panel to simulate the stamping action

Functional Requirements:

- If no part is on the conveyor it will not operate
- When a part is placed in front of the first sensor and the ready button is pressed, the part automatically moves to the second sensor
- When the part reaches the second sensor, the conveyor stops to allow the part to be stamped
- The part stamp can't function unless the conveyor has stopped and the part is in front of the second sensor
- After the part is stamped, it will automatically continue to the end
- After the part reaches the end, the conveyor stops and the system waits for a new part to be placed in front of the first sensor
- The conveyor motor can only be started 3 times within one minute to avoid overheating

Operating Sequence:

- Once program is started it waits for user to be ready
- Once the ready button is pressed, the program checks for a part on the conveyor in front of the first sensor
- If no part is found in front of the first sensor, the program displays the error and re-loops back to wait for the user
- If a part is found, the conveyor starts and the part begins moving down the conveyor
- The program displays to the user that the conveyor is moving
- When the part reaches the second sensor, the conveyor stops, the part gets stamped, and the part continues to the end of the conveyor
- The program waits until the part is removed from the conveyor
- If the stop button has not been pressed, the program re-loops

Documentation:

- VI online description
- Title information is presented on front panel
- Appropriate comments are provided on block diagram

Deliverables:

- Soft copy of the lab report
- Soft copies of VI and all SubVIs

Hardware Design for Project

The schematic for the hardware design is shown in Figure 2. To interface with the 24V DC motor, a motor driver circuit was designed that consists of a TIP31C transistor and a TIP32C transistor configured as switches. When the DAQ card outputs +5 V DC, both transistors are switched on. This will cause approximately 24V to be applied across the motor to turn it on. When the DAQ card outputs 0V DC, both transistors are switched off, which cause the motor to turn off.

To obtain a usable signal from the IR sensors, an LED driver circuit was designed that consists of a 3k ohm potentiometer to adjust the current passing through it in order to change the range of the detector. The detector circuit consists of a 1M ohm current limiting resistor, a 6k ohm base resistor connected to a 2n3904 transistor, and a 1k ohm base resistor connected to a 2n3906 transistor. When the detector sees sufficient infrared light reflecting off the part, both transistors are switched on. With the voltage divider from R₂ and R₃, this will cause 10V to be applied to the analog input channel of the DAQ card. When the detector does not see sufficient reflective infrared light, both transistors are switched off, which cause the analog input channel voltage to be 0V.

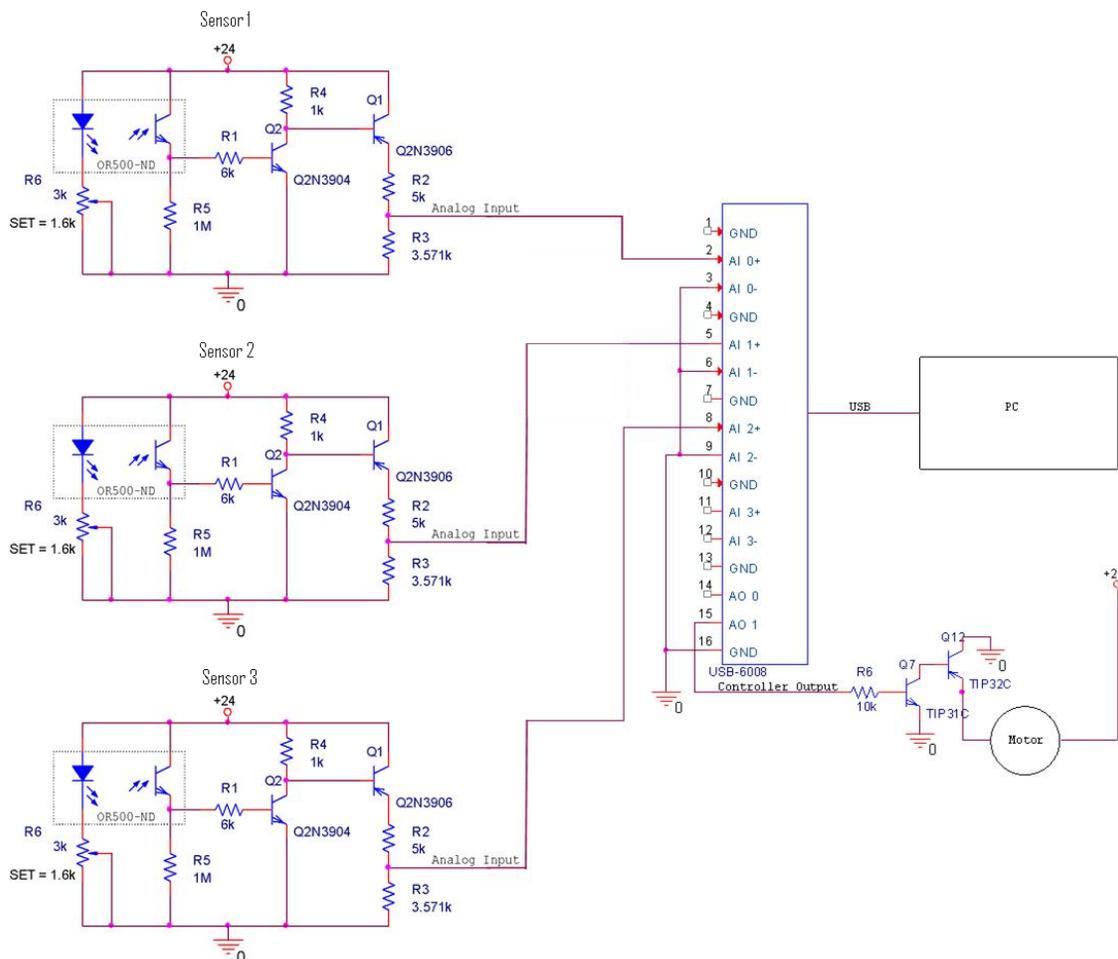


Figure 2. Schematic for PC Hardware Design.

Software Design for Project

A flowchart for the software design is shown in Figure 3. The flow of the program is as follows:

1. User presses the ready button.
2. The program then waits for a part to be placed in front of sensor 1.
3. When a part is placed in front of sensor 1, the program checks the number of times the conveyor motor has been started within 1 minute. If the conveyor motor has been started 3 times within 1 minute, the program will not allow the conveyor to start again until the 1 minute timer has timed out.
4. If the conveyor has not been started 3 times within 1 minute, the conveyor starts and the part moves toward sensor 2.
5. Once the part reaches sensor 2, the conveyor is stopped and the program waits for the stamp button to be pressed.
6. Once the stamp button is pressed, the stamp indicator is illuminated.
7. If the conveyor has not been started 3 times within 1 minute, the conveyor starts and the part moves toward sensor 3.
8. When the part reaches sensor 3, the program stops the conveyor operation, unlatches the ready button, and turns off stamp indicator.
9. If the stop button is not pressed, the program re-loops.

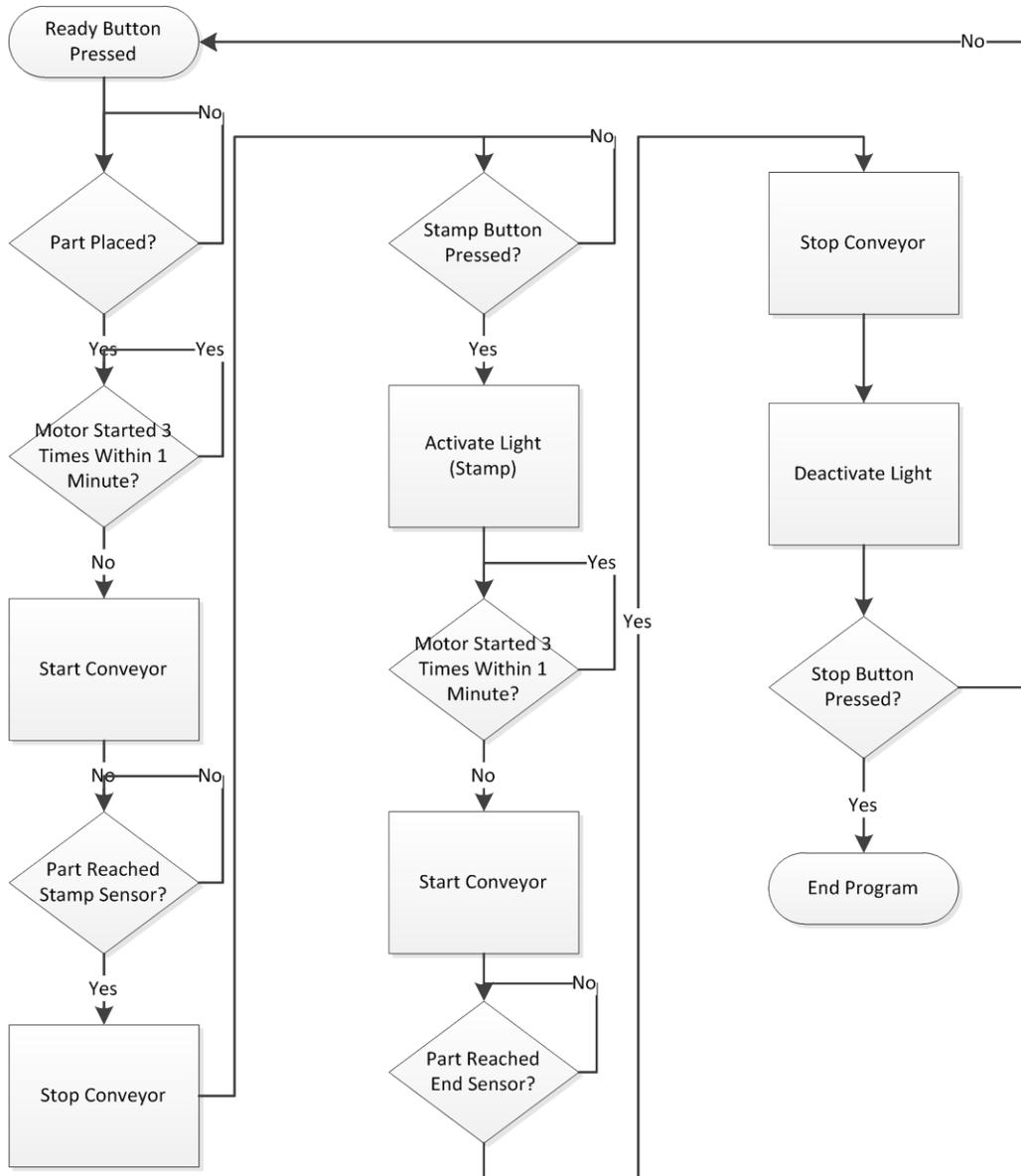


Figure 3. Flowchart for PC Software Design.

For the software design, a state machine was used. Table 1 lists each state of the state machine, along with the function of each state. Figure 4 shows the screen shot of a portion of the block diagram of the LabVIEW program.

Table 1. PC State Machine Design.

State	Function
Init	All variables are initialized to default, and it transfers to the “No Event” state.
No Event	Monitors all inputs (Sensor 1, Sensor 2, Sensor 3, Ready Button, and Stamp Button). When a state change occurs, it transfers to the appropriate state.
Run Conveyor	Starts the conveyor motor and transfers to the “No Event” state.
Sensor 2 Reached	Part has reached the middle sensor and the conveyor is stopped. It transfers to the “No Event” state and waits for the stamp button to be pressed.
Stamped	The stamp button has been pressed, and then the conveyor motor is started.
Sensor 3 Reached	Part has reached the last sensor and the conveyor is stopped. It transfers to the “No Event” state and waits for the stamp button to be pressed.

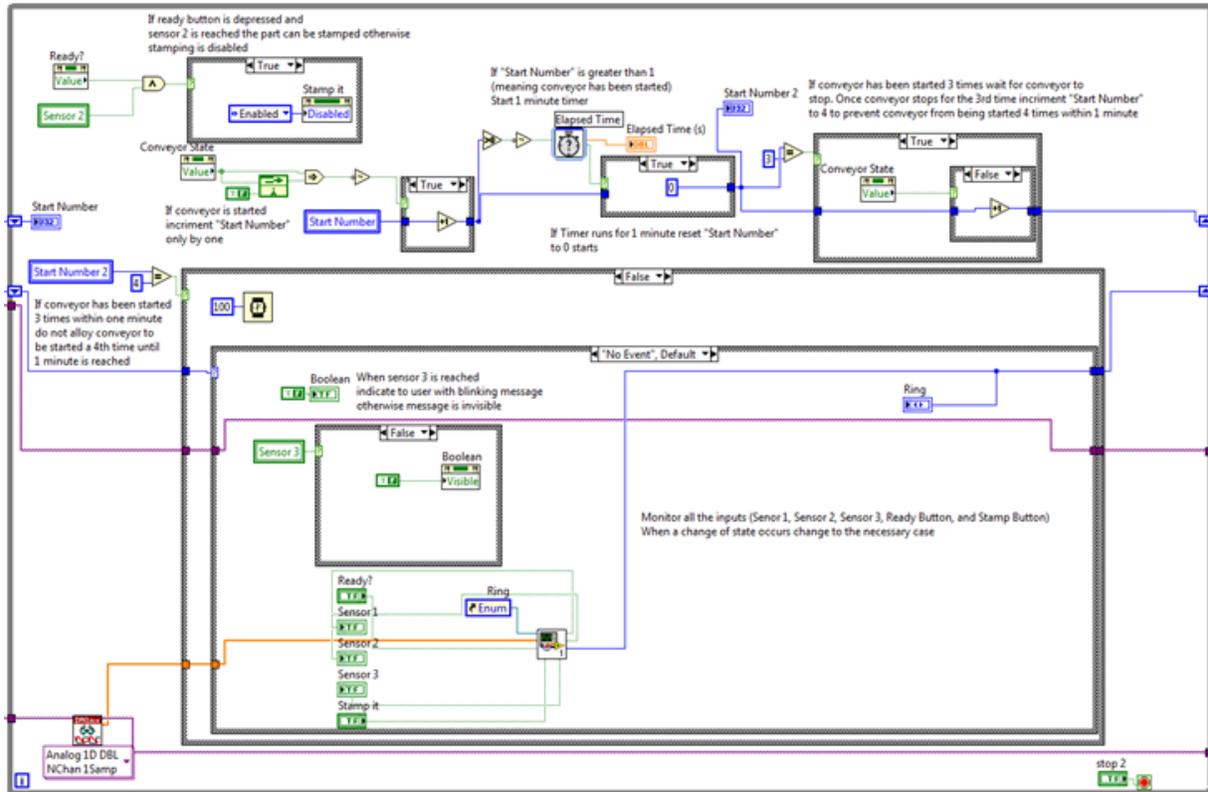


Figure 4. Screen shot of LabVIEW block diagram.

The screen shot of the front panel of the LabVIEW program is shown in Figure 5. It uses 5 Boolean indicators, 3 Boolean controls, 1 text indicator, and 2 channel inputs. The Ring indicator displays the current state of the state machine for debugging purposes. The sensor 1, 2, and 3 Boolean indicators show when a part is in front of each sensor. The conveyor state indicator displays the following:

- When the conveyor is running, the indicator is green and displays the message “RUNNING”.
- When the conveyor is not running, the indicator is red and displays “NOT RUNNING”.

The indicator directly above the conveyor state indicator is hidden unless the part reaches the third sensor. When it reaches this sensor, it becomes visible and blinks between the colors yellow and red and displays “PLEASE REMOVE PART”. The button that displays “READY” is used for the user to tell the program when they are ready to start operation. The “Stamp it” control is used for the user to “Stamp” the part. This control is disabled unless the part is in front of sensor 2 and the conveyor is stopped. The “STOP” button is used to end the program. The “Input Channels” control is used to specify the input channels for the three sensors. The “Output Channel” control is used to select the channel that will interface to the motor control circuit which runs the conveyor.

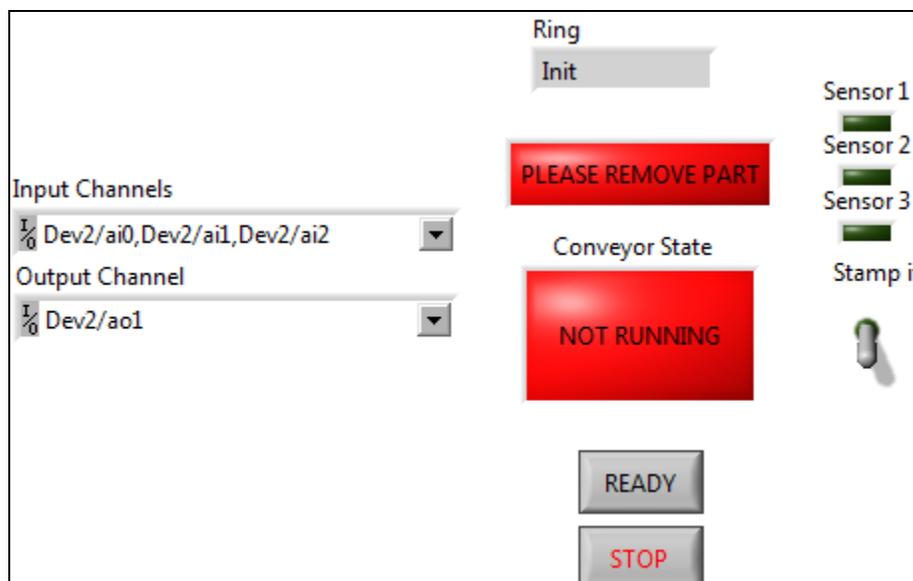


Figure 5. Screen shot of front panel of LabVIEW program.

Functional Tests for Project

Several tests were performed to demonstrate successful operation of the elevator control system, as shown below in Table 2. Each test is numbered with a description and a pass/fail indication.

Table 2. Functional test procedure.

Test number	Function	Description	Pass/Fail
1	Ready Button	Conveyor button does not operate unless the ready button is latched.	
2	Starting the conveyor	Conveyor and part on the conveyor will start moving only if the ready button is pressed and the part is in front of sensor 1.	
3	Part reaches sensor 2	Conveyor motor stops when part reaches sensor 2.	
4	Stamp button	Part will not continue until stamp button is pressed. Once stamp button is pressed, the conveyor starts and part continues down conveyor.	
5	Sensor 3 reached	When sensor 3 is reached, conveyor stops operating and user is notified.	
6	Remove part and place at beginning	The process repeats when part is removed from sensor 3 and replaced at sensor 1.	
7	Ready button is unlatched during operation	Conveyor stops immediately and notifies user.	
8	Overheating	The conveyor motor can only be started three times within 1 minute to avoid overheating.	

PLC Implementation

The PLC implementation was performed on an Allen-Bradley controller⁷. The parts list is shown in Table 3 and the schematic is shown in Figure 6. Some additional functional requirements were specified for the PLC design.

- If a 10-second timer times out before the part reaches the second sensor, the conveyor is stopped and an alarm is lit. The program will only continue after an alarm acknowledge button is pressed.
- A 5-second timer is used to simulate that a part has been stamped.

These additional requirements were specified by the instructor for the sophomore-level PLC course.

Table 3. Parts List for PLC Implementation

Controller	1769-L32E
Power Supply	1769-PA2
Digital Input	1769-IQ16
Digital Output	1769-OW16
Ethernet Switch	---
2 NC Push Buttons	Run, Alarm Acknowledge
1 DC Motor	Motor
2 Lights	Motor Running, Alarm
3 IR Proximity Sensors	OR500-ND

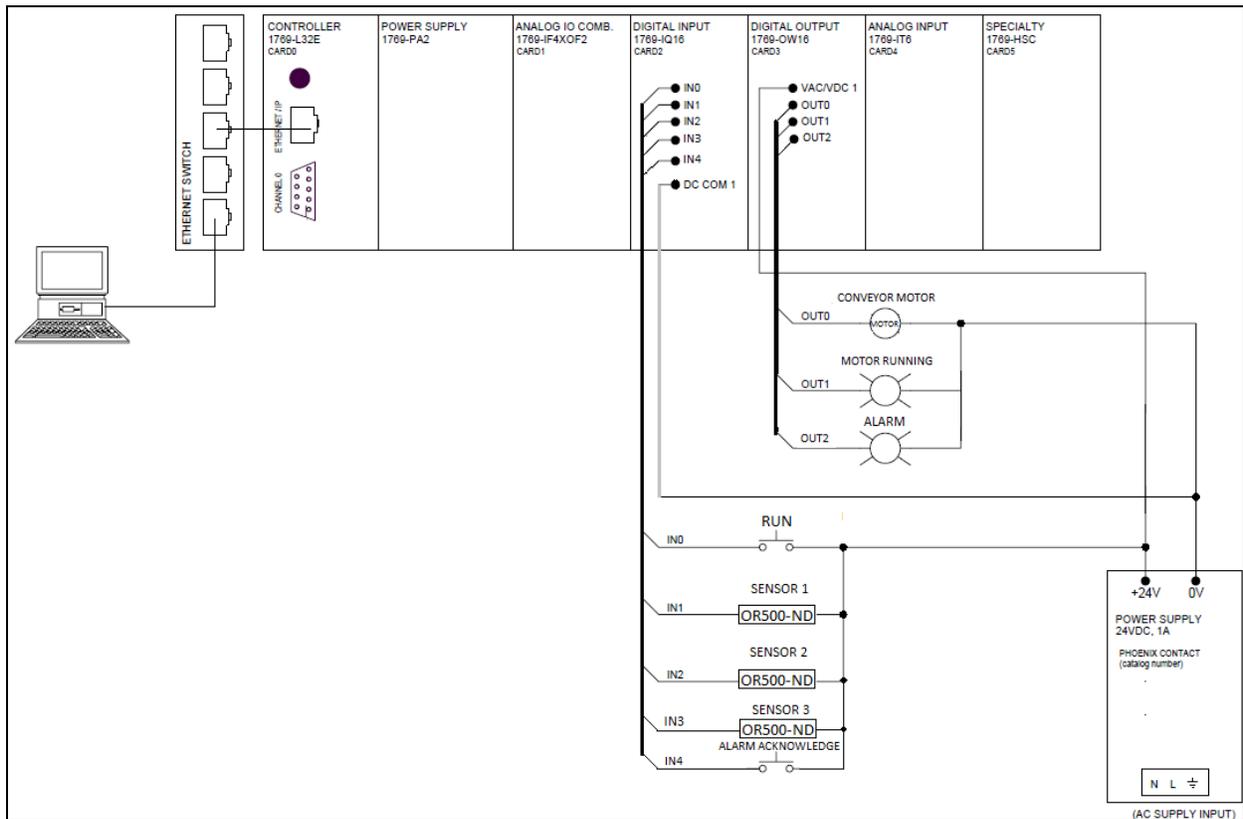


Figure 6. PLC schematic.

Table 4 lists each rung of the ladder diagram along with its function. Figure 7 shows the screen shot of the ladder diagram.

Table 4. Ladder Diagram Design.

Rung	Function
0	If ready button becomes unlatched, move the value 0 to the motor_started tag and unlatch motor output and stop conveyor.
1	If ready button is true, 10 second timer has not timed out, and motor output (Start) is true, then move the value 1 to the Motor_Minute tag, increment the number of times the motor has been started by 1, and illuminate the Motor_Running_Indicator.
2	If the motor has been started 3 times within one minute, wait for the motor running indicator to become false. Once it becomes false, move the value 4 to the number of times the motor has been started within 1 minute. This stops the conveyor from being started a 4 th time within 1 minute.
3	If the motor has been started 3 times within a minute, keep the motor output false.
4	Once the 1 minute timer has timed out, reset the number of times the motor has been started and the 1 minute timer.
5	If ready switch is true, the motor has not been started more than 3 times within 1 minute, and the part is in front of sensor 1, jump to the start motor subroutine.
6	If the ready switch is true and the motor has been started, then start the 10 second timer.
7	If the ready switch is true and the motor has been started, start the 1 minute timer.
8	If the 10 second timer has not timed out and the part reaches sensor 2, jump to the Sensor_2_Reached subroutine.
9	If the 10 second timer times out before the part reaches sensor 2, stop the conveyor and illuminate the alarm. Once the alarm acknowledge button has been pressed, reset the 10 second timer.
10	If sensor 3 is reached while the conveyor is running, jump to the Sensor_3_Reached subroutine.

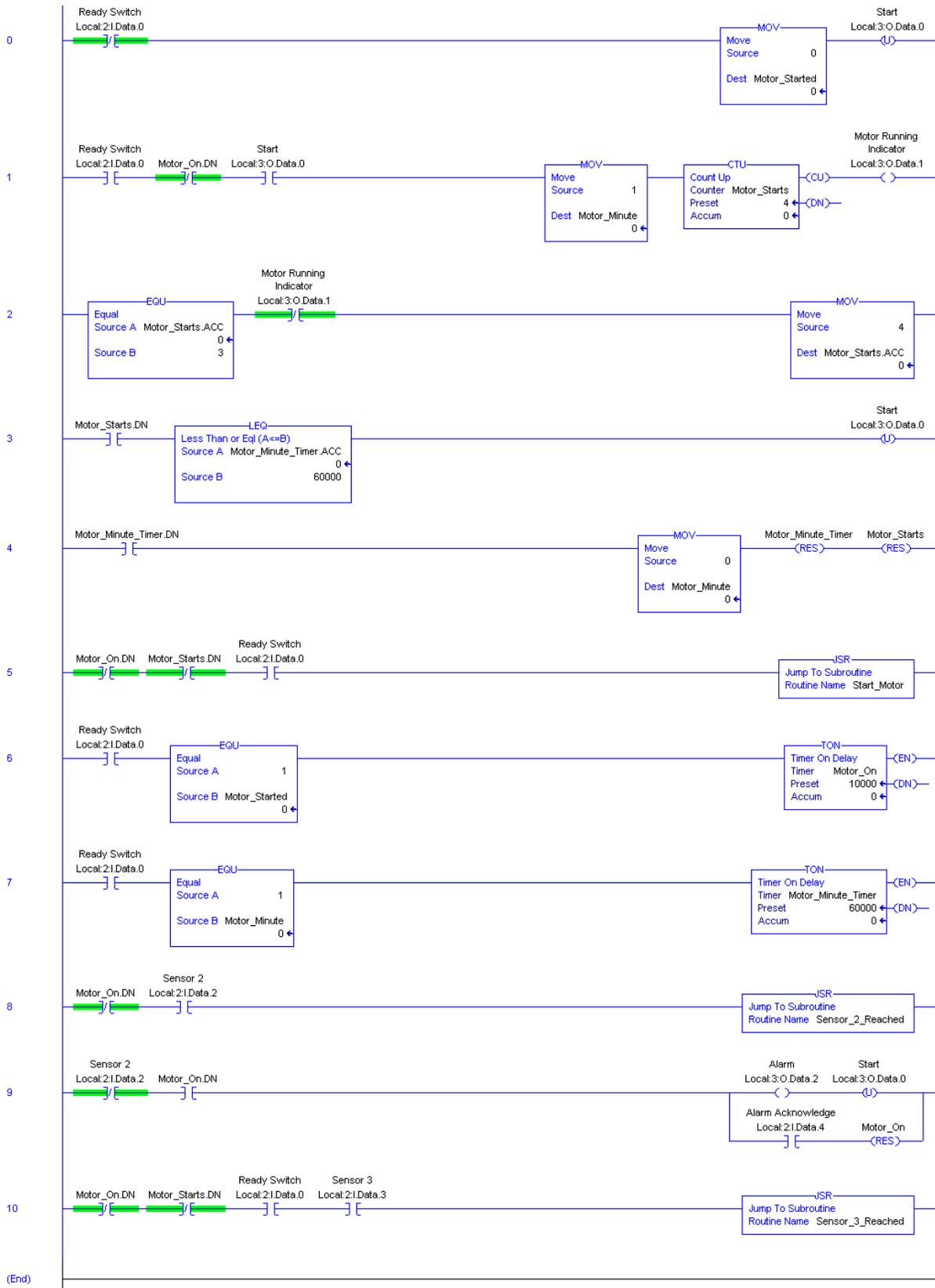


Figure 7. Screen shot of ladder diagram.

Project Assessment

Table 5 provides results from a student questionnaire. Table 6 provides a student comparison of the advantages and disadvantages for the PC and PLC systems. Overall, the student response to this project indicates that it was a great learning experience. However, more data will need to be collected for statistically relevant results.

Table 5. Student Questionnaire Results.

Questions	PC System	PLC System
Approximate time in hours to complete hardware design	5	5
Approximate time in hours to complete software design	5	6
Overall level of difficulty	Low	Low
Areas of difficulty	Hardware, State Machine	Timers, Counter
Suggestions for improvement to help ensure student success	Review of hardware interfacing electronics	Review of timers and counters
Suggestions for number of students on this project	2	2
Suggestions for number of weeks to complete project	2	2

Table 6. Student Comparison of PC and PLC Systems.

System	Advantages	Disadvantages
PC	<ul style="list-style-type: none"> • Since the PC system used the USB DAQ, it was much more portable than the PLC system. • All of the controls and indicators for the PC system were contained on the front panel. 	<ul style="list-style-type: none"> • The PC system is not as robust as the PLC system and requires a working PC running specific software to interface with the specific hardware. • The system also is not a real-time control system. This was not a problem due to the slow time response characteristics for this system.
PLC	<ul style="list-style-type: none"> • The PLC system did not need any separate circuitry to run the conveyor's DC motor. • The inputs and outputs are very easily addressable and require no extra programming to separate the incoming data. • The system does not need a PC connected to it once it is programmed. 	<ul style="list-style-type: none"> • The PLC system was too large to be portable. • It had to be tested in the lab. • It required having separate pushbuttons and indicators as well as the controller and input cards. • It required more hardware wiring. • The Allen Bradley controller and software are much more expensive than the PC hardware and software.

Conclusions

In this paper, results are shown for both a PC-based and PLC-based conveyor control system. Due to the slow time response characteristics of this system, the PC-based control system has been as effective as the PLC system. Both types of systems integrate hardware and software designs. The tradeoffs to consider are real-time control, portability, robustness, and cost.

Recommendations

For the PC system, it is recommended that transistor theory be reviewed prior to the start of the project. Also, 2-3 hours of lecture should be allotted for introducing state machines. This should also include an example of the design of a state machine.

For the PLC system, it is recommended to review how timers and counters function. Students will also need to be familiar with how NO and NC contacts work both in hardware and software.

Acknowledgment

I would like to thank Roger Kuntz, the instructor for the sophomore-level PLC course, for providing information regarding the engineering requirements for the PLC implementation of this project.

Bibliography

1. Loker, D., "Elevator Control System Project," *Proceedings of the Annual Meeting, American Society for Engineering Education*, 847_ELEVATOR_CONTROL_SYSTEM_PROJECT.pdf on CD-ROM, 10 pp., Louisville, KY, 2010.
2. Bishop, Robert H., *Learning with LabVIEW 2009*, Pearson Education, 2010.
3. Travis, Jeffrey and Jim Kring, *LabVIEW for Everyone*, 3rd Edition, Pearson Education, 2007.
4. Nise, Norman S., *Control Systems Engineering*, 5th Edition, Wiley, 2007.
5. Web Site <http://www.digikey.com>.
6. Web Site <http://www.ni.com>.
7. Web Site <http://www.ab.com>.