MSE 310 - Introduction to Electro-Mechanical Sensors and Actuators

Project Title: Industrial Control Trainer (ICT) Management via LabVIEW
Project deadline:

Sepehr Rezvani - 301291960
[redacted]

Table of Contents

Introduction **2**

Section 1. Industrial Control Technology 2

PartA- Sorting Area 4

Part B- Assembly Chute 6

Part C- Sensing Station 7

Part D- Reject Area 9

Section 2 NI-MAX 10

Section 3. LabVIEW Code 12

Conclusions **16**

References **17**

Introduction

In this project, Industrial Control Technology unit (ICT3) sensors and actuators were controlled with LabVIEW code that was written by our group. Note that the components of ICT cannot be directly controlled by software on a computer. Therefore, a PC-Based Data Acquisition Interface (DAQ) was required to read sensors' data from ICT (DAQ input), and write on an ICT actuators (DAQ output). NI USB-6501 was the DAQ used for this project and NI-MAX was its graphical user interface.
A full description of LabVIEW, NI-MAX and ICT performance and mechanism are included in the following sections of this report.

The main objective of this project was to write a code in LabVIEW that receives data from ICT's sensors, analyzes the information and controls actuators on the ICT. LabVIEW played the major role in this project and ICT can be turned on/off from the Front Panel of LabVIEW. Additionally, the status of all sensors and actuators can be viewed from LabVIEW's Front Panel (whether they are on or off).

Lastly, the code that was written in LabVIEW to control ICT has two modes, one to run the system automatically (start ICT, place plastic rings or metal pegs on the Chain Conveyor, and wait until all the pegs/rings are sorted), and the other mode is to control each step manually. A scenario that was considered for this project that was very relevant to real-life situations was power outage. In this case, the condition of all sensors/actuators on ICT must have been saved and as soon as power came back, ICT must continue.

Research and Analysis

This section includes full description of different components of the project and how they communicate together are included.

Section 1. Industrial Control Technology

ICT is a representation of an industrial automation system that includes component sorting (metal peg versus plastic ring), assembly (assembling ring on top of peg) and accept/ reject process (combination of sensors to detect and actuators to push objects). To start-up ICT, the following should be performed in order:

1. Make sure the Emergency Stop Button (Figure 1-G) has been released
2. Turn computer on

3. Press start from either LabVIEW's Front Panel or Start Button on ICT (Figure 1- E)
4. Switch the DC power supply: 4 Volts- 12 Volts
5. Connect the AC plug

To turn off ICT

1. Disconnect AC power
2. Turn off DC power supply
3. Press the Stop Button on ICT (figure 1-F) or stop icon on LabVIEW's Front Panel
4. Turn computer off

Note: Emergency Stop Button is designed to stop ICT's process in case of an emergency only and when pressed, the controller is stopped and the supply voltage to all actuators is cut. So, use either one of the options in step 3 above in order to stop the process.

ICT also includes Belt Conveyor (Figure 1-H), Chain Conveyor (Figure 1-I), 9 sensors and 5 actuators that are positioned in multiple stations (Figure 1, A-D). Priority of ICT is to assemble plastic rings on top of metal pegs by the time they reach the end of Belt Conveyor. For many reasons, this goal might not be accomplished. In that case, code written in LabVIEW would be able to distinguish rings or pegs that are not assembled, and push either down a separate bin (by the actuator located in section D). In general, sensors and actuators have the following role in the process:

Sensors: sense movement of objects (peg/ring) or physical components of ICT

Actuators: Push objects (peg/ring) off the Chain/Belt Conveyor or move physical components of ICT.

Note: These are general roles of sensors/actuator. Detailed description of them will be discussed in the next section.

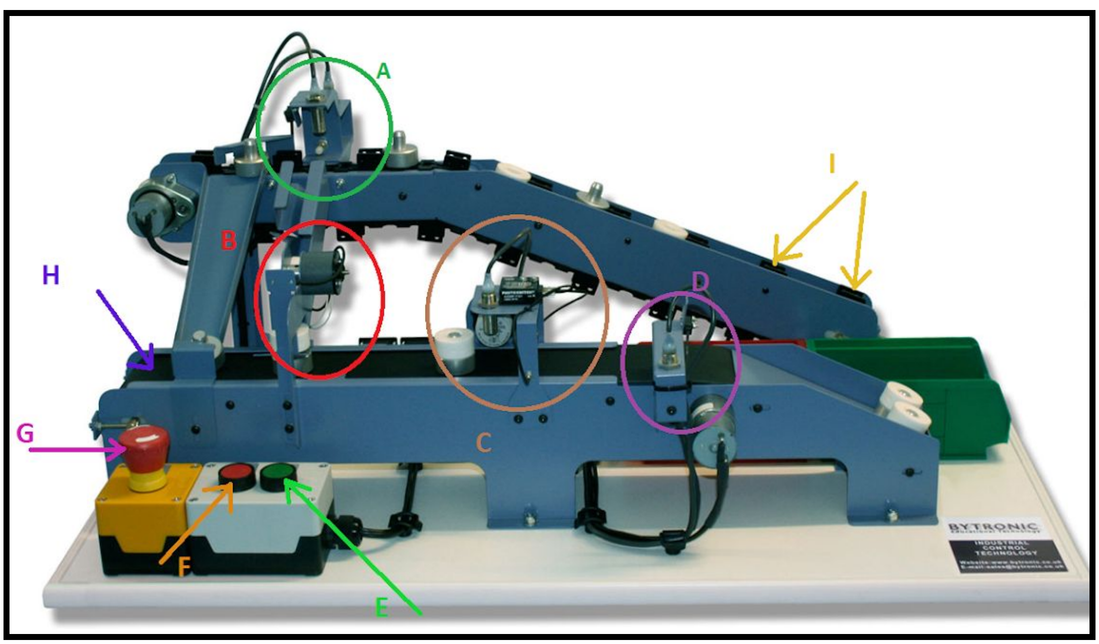

Figure 1.1

PartA- Sorting Area

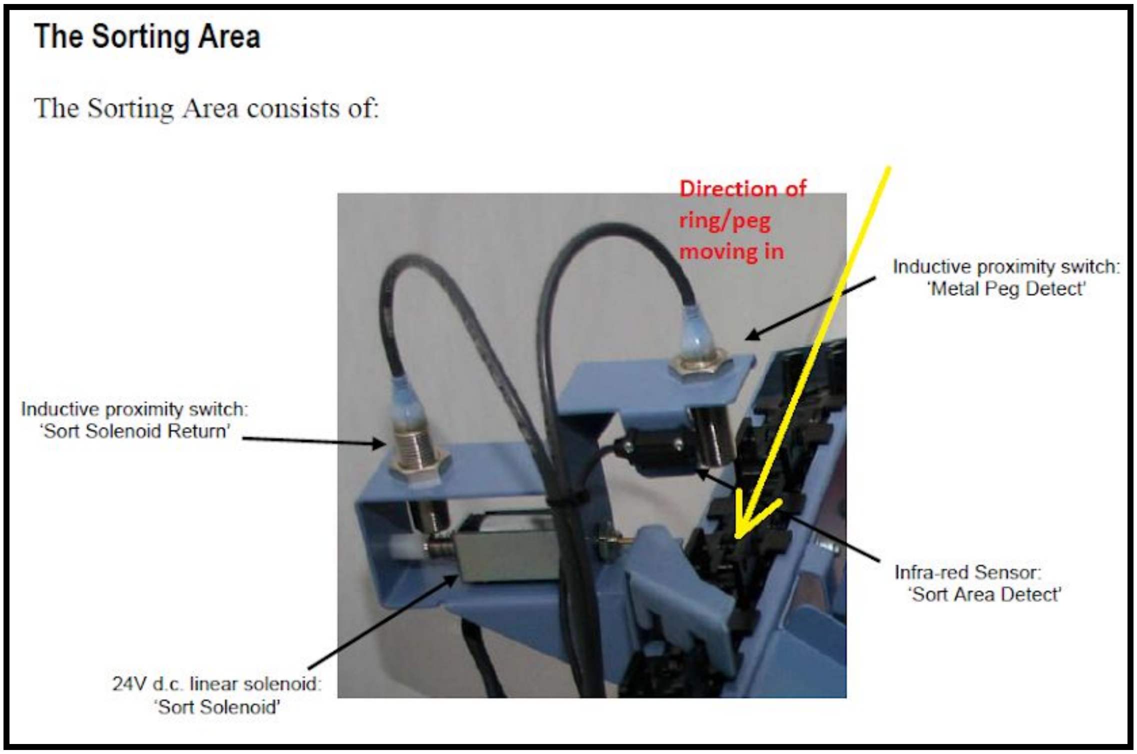

Figure 1.2

Components of this part A are labeled in Figure 2 and each of sensors/actuators are connected to a pin on NI USB-6501. Table 1 describes the components of this part, their type, role, pin number and normal state.

Name	**Type (Sensor/Actuator)**	**Role**	**Pin number on NI USB-6501**	**Normal state (0/1)**
Sort Area Detect	Sensor	Detects ring/peg entering the area	P0.2	0
Metal Peg Detect	Sensor	Detects metal peg only	P0.3	0
Sort Solenoid	Sensor	Monitors 'Sort	P0.4	0

Return		Solenoid’		
Sort Solenoid	Actuator	Pushes plastic ring off of Chain Conveyor	P2.1	0

Table 1.1

Process:

1. Object (ring/peg)enters area
2. First detected by ‘Sort Area Detect’
3. ‘Sort Area Detect’ is positioned in a way that it only detects when object is directly under ‘Metal Peg Detect’
    3.1. If ‘Metal Peg Detect’ is activated before ‘Sort Area Detect’, LabVIEW code decides that object is peg and does not activate ‘Sort Solenoid’ actuator (allows the object to pass).
    3.2. Otherwise, if only ‘Sort Area Detect’ is activated, code decides that object is plastic ring and activates ‘Sort Solenoid’ actuator.
    3.3. If the object was metal peg, it continues its journey on the Chain Conveyor, gets deflected at the end of the track and falls down on the Belt Conveyor.
    3.4. If object was plastic ring, it gets off Chain Conveyor after getting hit by ‘Sort Solenoid’. At this point, the object its on the way to Part B
4. ‘Sort Solenoid Return’ monitors performance of ‘Sort Solenoid’

Part B- Assembly Chute

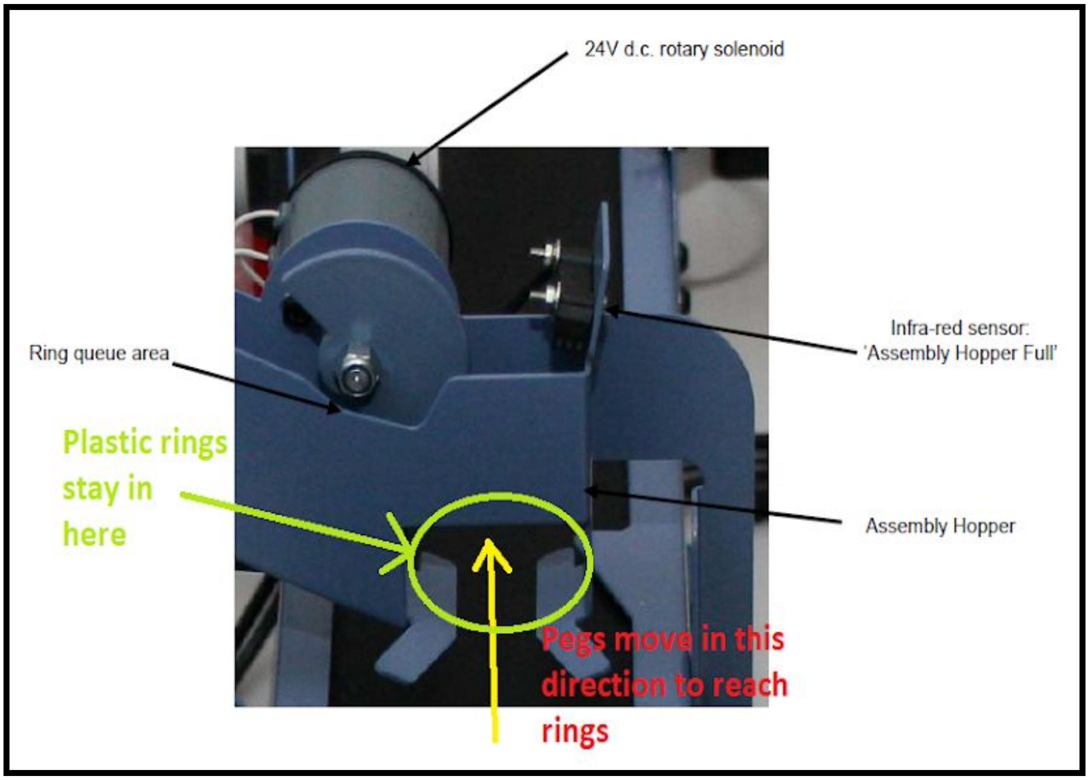

Figure 1.3

Components of this part B are labeled in Figure 3 and each of sensors/actuators are connected to pins on NI USB-6501. Table 2 describes the components of this part, their type, role, pin number and normal state.

Name	Type (Sensor/Actuator)	Role	Pin number on NI USB-6501	Normal state (0/1)
Assembly Hopper Full	Sensor	Checks if Assembly Hopper is empty/full	P0.5	1
Rotary Solenoid	Actuator	Rotates to release a single ring	P2.2	0

Table 1.2

Process:

1. Following the part A process, plastic rings and metal pegs were separated before entering part B.
    1.1. Rings were pushed down by ‘Sort Solenoid’ down to ‘Ring Queue Area’ (Figure 3)
    1.2. Pegs moved to end of Chain Conveyor and deflected down to Belt Conveyor
2. Based on LabVIEW code, if ‘Sort Solenoid Return’ is 1 (‘Sort Solenoid’ actuator was activated) logically there will be at least one ring in ‘Ring Queue Area’. Therefore, ‘Rotary Solenoid’ is activated and releases one ring to Assembly Hopper (Figure 3)
3. Plastic ring will wait in Assembly Hopper until a metal peg moves to this area and will be loaded on the peg. Note that this is a purely physical process and is independent of any sensor, actuator or computer code.

Note: The controller keeps track of the number of rings in queue area (area between part A and ‘Assembly Hopper’). Therefore, queue counter increments or decrements when a ring is pushed down from part A to B, or when ‘Rotary Solenoid’ is activated, respectively. This process will be described more specifically in LabVIEW section.

Part C- Sensing Station

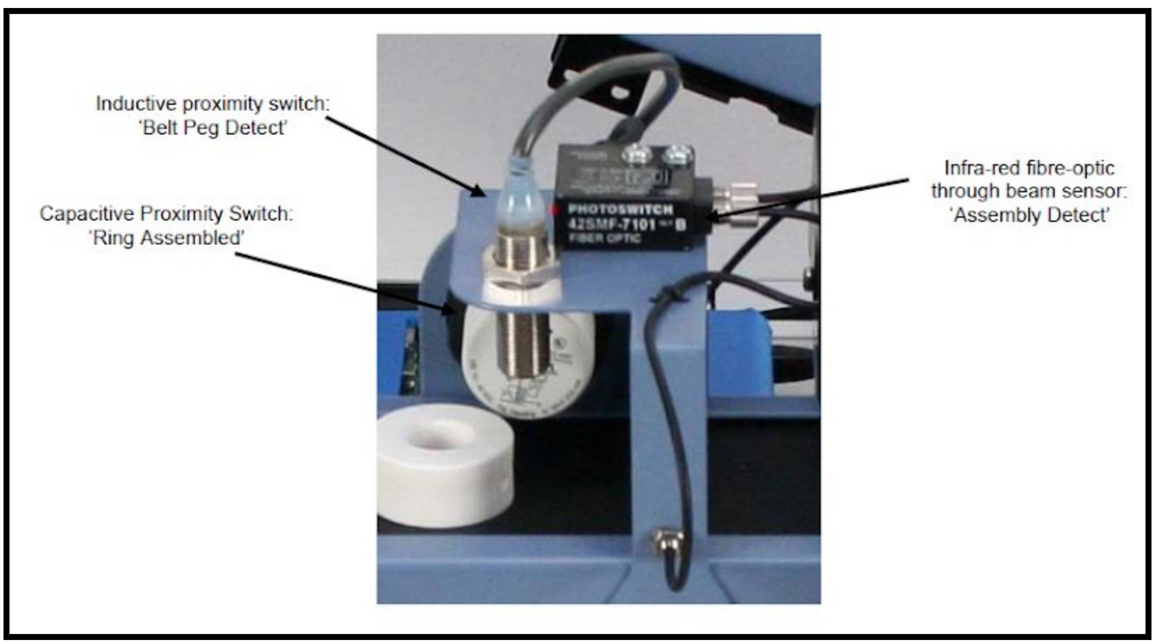

Figure 1.4

Components of this part C are labeled in Figure 4 and each of sensors/actuators are connected to pins on NI USB-6501. Table 3 describes the components of this part, their type, role, pin number and normal state.

Name	Type (Sensor/Actuator)	Role	Pin number on NI USB-6501	Normal state (0/1)
Belt Peg Detect	Sensor	Activated when it sees metal peg or assembled (ring on peg)	P0.6	0
Ring Assembled	Sensor	Activated ONLY when assembled (ring on peg)	P0.7	1
Assembly Detect	Sensor	Activated when it sees either one of the 3 options (ring, peg, or both)	P1.0	1

Table 1.3

Process:

1. As any objects enter part C, there are three possibilities
    1.1. Only ‘Belt Peg Detect’ and ‘Assembly Detect’ are activated. Therefore, ‘Ring Assembled’ is not activated and it means the object passing by is a metal peg.
    1.2. Only ‘Assembly Detect’ is activated. Since the other two sensors are deactive, it is concluded that object passing is neither assembled nor a metal peg. Therefore, it is a plastic ring.
    1.3. All three sensors are active. This means the object passing is fully assembled (ring on peg).

In each of these three scenarios (1.1-1.3), the data from sensors will be collected and used later in part D.

Part D- Reject Area

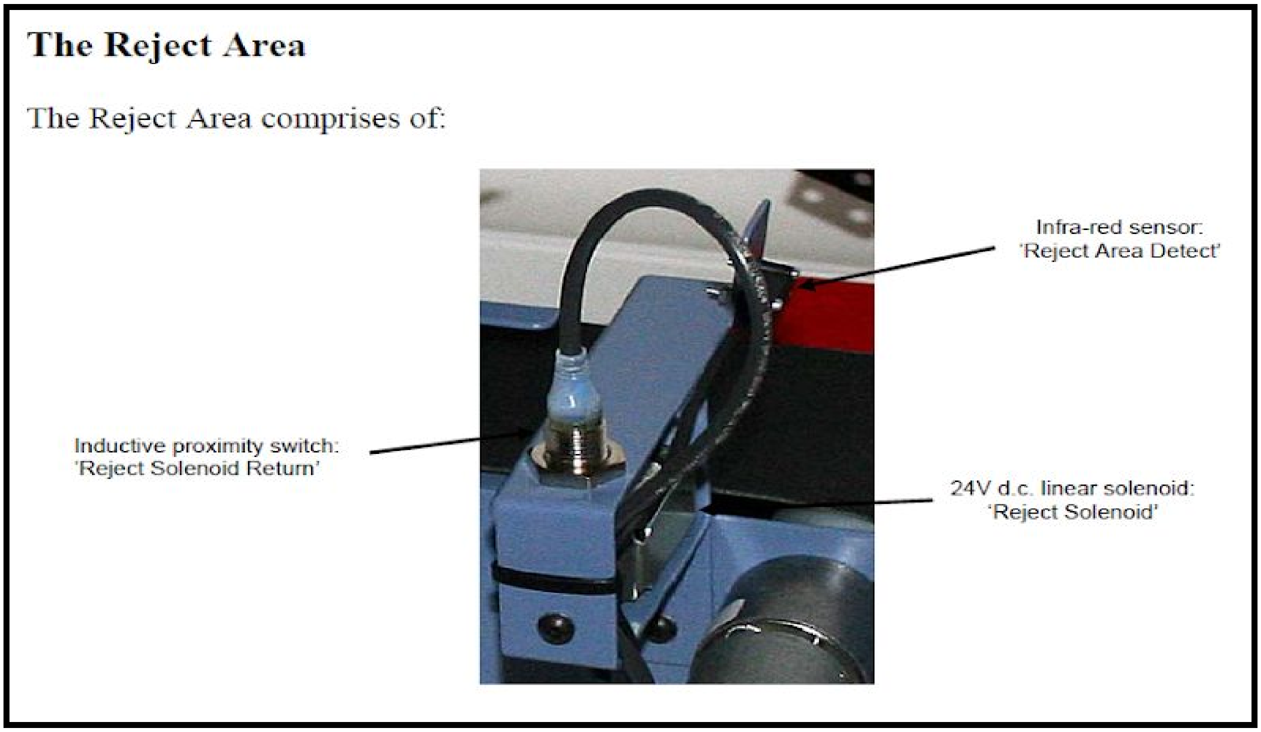

Figure 1.5

Components of this part C are labeled in Figure 5 and each of sensors/actuators are connected to pins on NI USB-6501. Table 4 describes the components of this part, their type, role, pin number and normal state.

Name	Type (Sensor/Actuator)	Role	Pin number on NI USB-6501	Normal state (0/1)
Reject Area Detect	Sensor	Detects any component entering	P1.1	0
Reject Solenoid Return	Sensor	Monitors 'Reject Solenoid'	P1.2	1
Reject Solenoid	Actuator	Activated when object either a	P2.4	0

		single peg or ring into the bin		

Table 1.4

Process:

1. As any object enters part D, ‘Reject Area Detect’ detects it. This sensor is positioned in a way that can sense objects only when they are exactly in front of ‘Reject Solenoid’
2. The counter has saved identity of objects on Belt Conveyor from part C and there are two possible outcomes in this part
   2.1. If object is either a single peg or a single ring, ‘Reject Solenoid’ will be activated to collect unassembled components in a bin
   2.2. If object entering is an assembled plastic ring on metal peg, ‘Reject Solenoid’ will not be activated. Therefore, all the assembled members will stay on Belt Conveyor and reach the end (figure 1).
3. ‘Reject Solenoid Return’ monitor records activation of ‘Reject Solenoid’ and sends the data back.

Section 2 NI-MAX

National Instruments (NI) provides Measurement and Automation Explorer (MAX) is a graphical user interface. NI-MAX is installed with one of the NI application development environments(LabVIEW), and with one of our hardware product drivers (NI USB-6501).

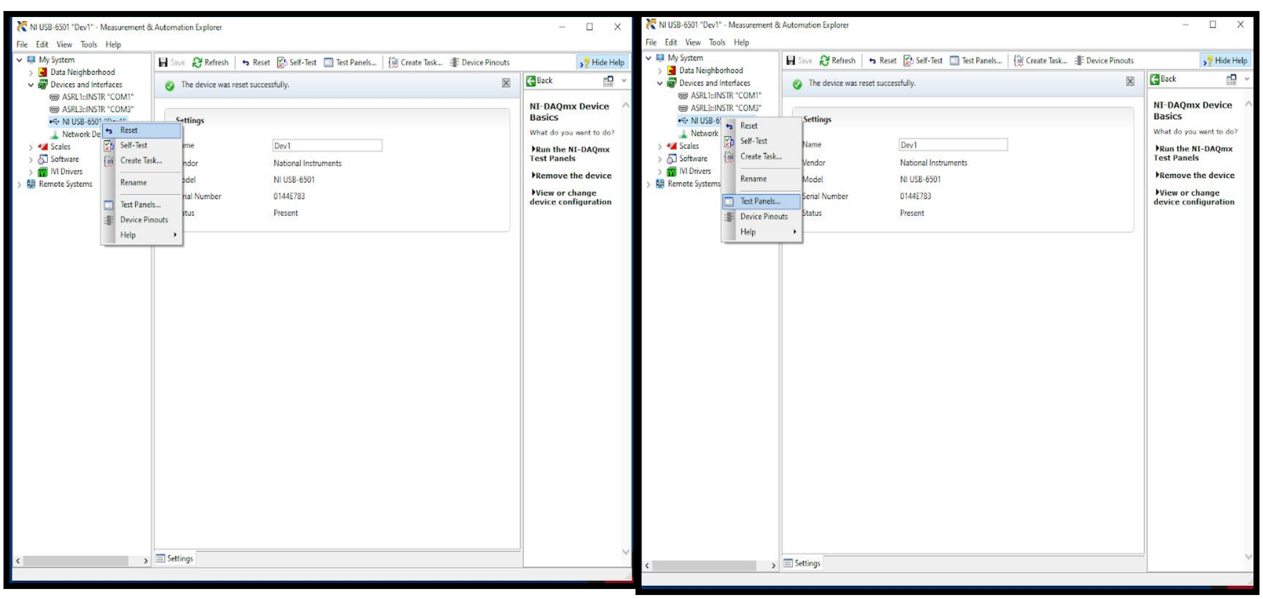

Figure 2.1

As mentioned in ICT section, sensor and actuator are connected to pins on the 'NI USB-6501'. NI-MAX software helped us understand the specific pin connection of sensors and actuators. In order to see pin connections, the following instructions must be followed:

1. Open NI-MAX software.
2. Select 'Devices and Interfaces'
3. Select NI-USB
    3.1. Right-click and select 'reset' → Figure 2.1
    3.2. Right-click again, select 'Test Panels' → Figure 2.1

At this point, either of ports 0, 1 or 2 can be selected. Ports 0 and 1 include sensors and port 2 actuators (Figure 2.2).

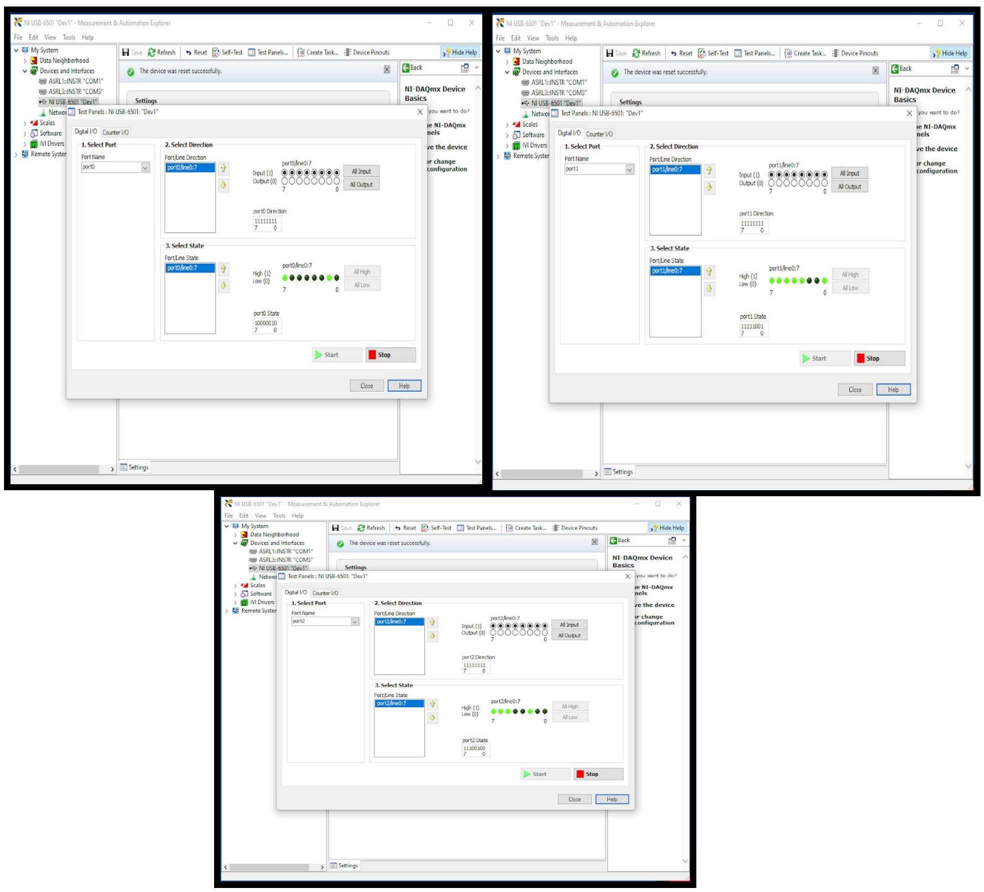

Figure 2.2

Note: Right two columns on tables 1-4 were indicated by inspecting changes in pin states on NI-MAX. For example, to indicate which pin belongs to 'Metal Peg Detect' from Section A, a peg was positioned underneath that sensor and whichever LED light on MAX turned on/off, belonged to that sensor. Same process was followed to indicate pin connections of ICT sensors and actuators.

Section 3. LabVIEW Code

We have divided the main functionalities of the project in parts. The bulk of the project lies in the automatic functions of the ICT. Thus, we have divided the project in locations of interest in the machine. Part A, **sorting area**, consists of the first set of sensors and actuators along the chain belt. Part B, **assembly chute**, mainly deals with the ring hopper and its sensors. Part C, **sensing station**, is where the capacitive and inductive sensor that differentiates between a complete object and an incomplete one. Finally, part D, **reject area**, consists of the last sensor and actuator that denies or accepts objects.

The primary method that we used to implement the code is edge detection and variable flags that are not directly connected to the ICT. That is implemented using case structures. It is clearly shown in Figure 3.1 for the code of the start/stop buttons, as well as the conveyor activation buttons. Some of the sensor values have a '1' as their normal state. So for simplicity, we had set all those machine values as 0 using not gates.

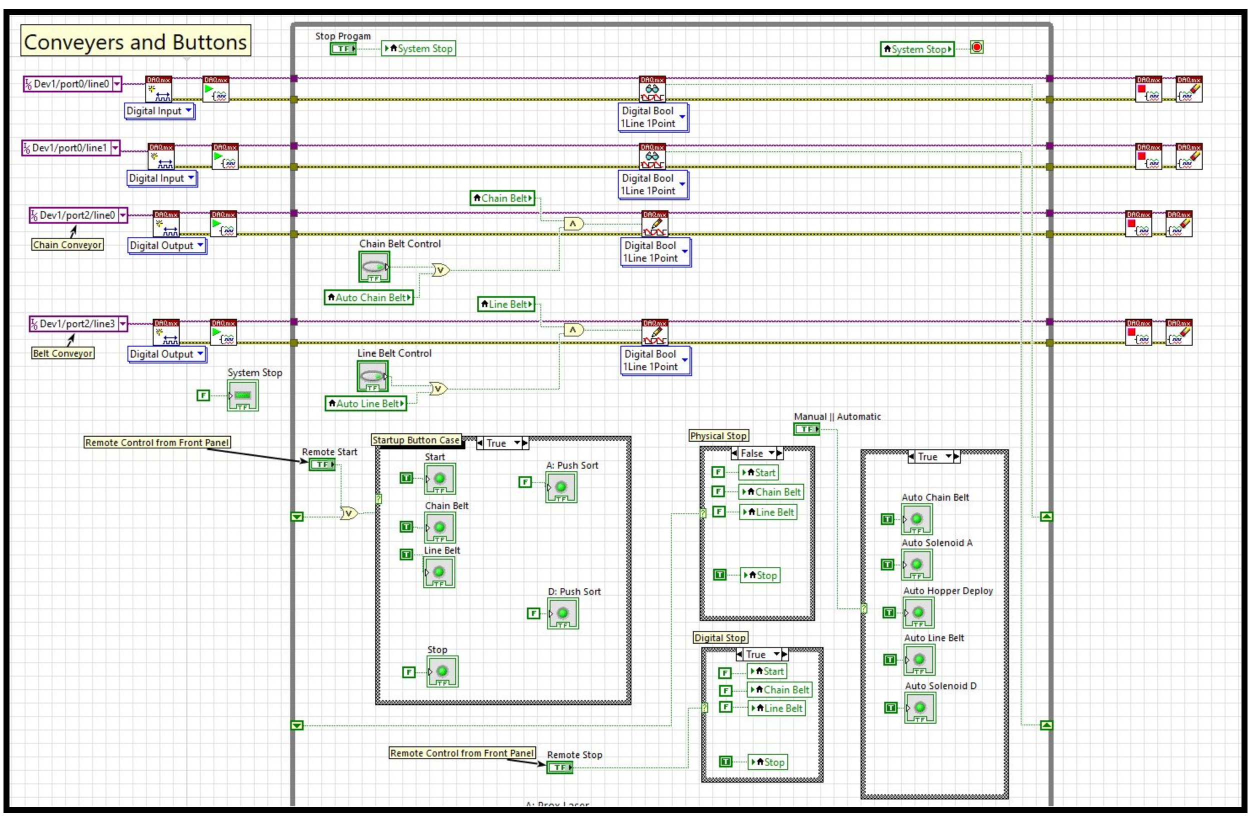

Figure 3.1

Part A consists of the primary sorting area between ring and metal. The core function of this part is to kick when the sensors don't detect a metal, and do not kick when it does sense a metal. With that in mind, we can solely use the proximity laser sensor in this part to be the sensor that detects an object in front of it. Thus, as a baseline, we can set that if the proximity laser senses something in front, then the push A solenoid will be activated. A flag variable is used “A: Metal Var” to deactivate that functionality once. That variable will reset once an object leaves the proximity laser. That is implemented in the form of a boolean crossing - where the proximity lazer senses a falling edge.

Another notable function in part A is that to not push rings if there are 5 in the hopper The method of incrementing and decrementing numbers is done the same way for every part. When the “A: Metal Var” is not activated AND the proximity sensor senses an object, we activate the flag “A: Push Sort”. When it is sent to a case structure, that True will activate another flag “A: Push Var” and will increase the number of items in the queue. In other words, that number will increment when the solenoid returns to resting position (negative edge).

The other functions that exist in the figure below refers to the manual mode of the system. That is done simply through logic gates. We made sure that the counters also worked in manual mode. This is all assuming that the user activates the push sort in the right times. That part A actuator will not work if the user performs incorrect movements such that the ICT senses the hopper is full

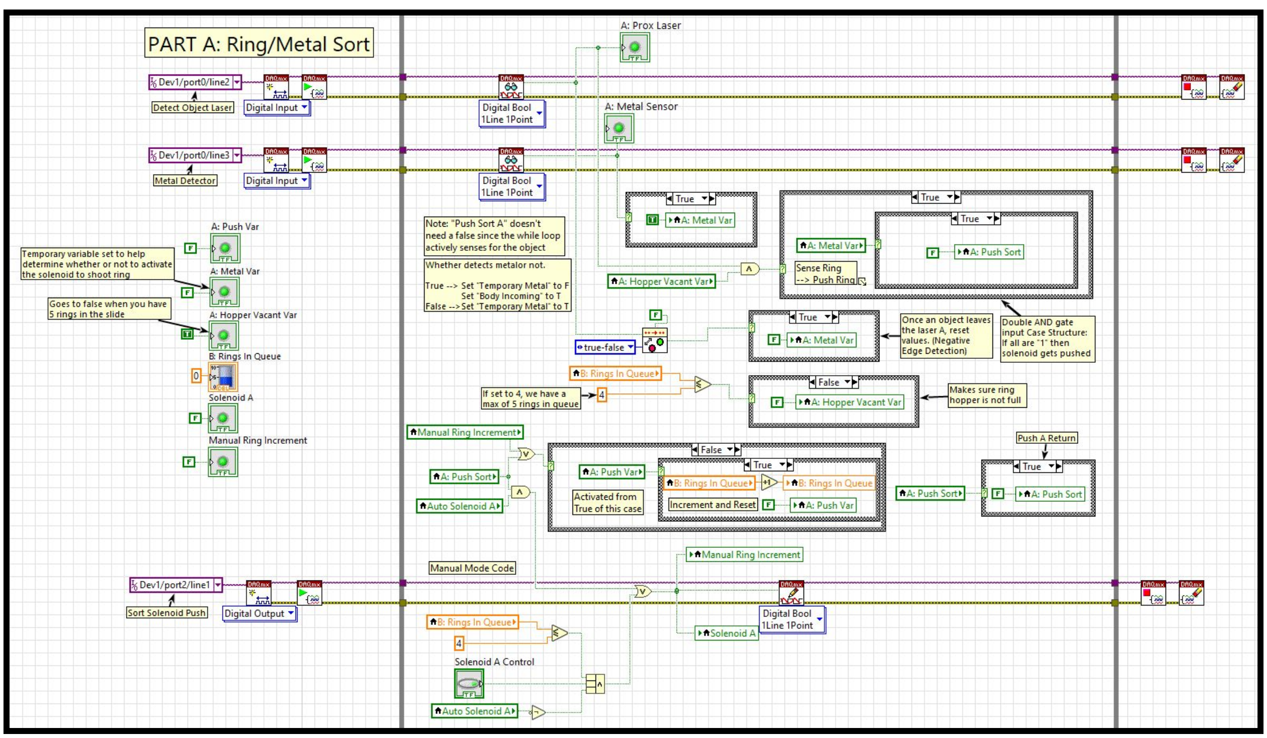

Figure 3.2

Functionalities like incrementing/decrementing numbers and manual mode is applied in the same way for each part. Part B only deals with the assembly chute, where the ring is combined with the metal. This case, there is only one sensor in this part, a sensor if the platform is empty. If there is nothing there, you deploy the next ring from the hopper. The most notable portion of this part of the code as seen in figure 3.3 is that it has its own while loop. That is because the time it takes for the first ring to fall into the front of the platform cannot function without a “delay”. Delays shuts down the functionality of the LABView code and essentially makes every sensor blind and actuator frozen for that set amount of time which in this case is 750ms. The rest of the code

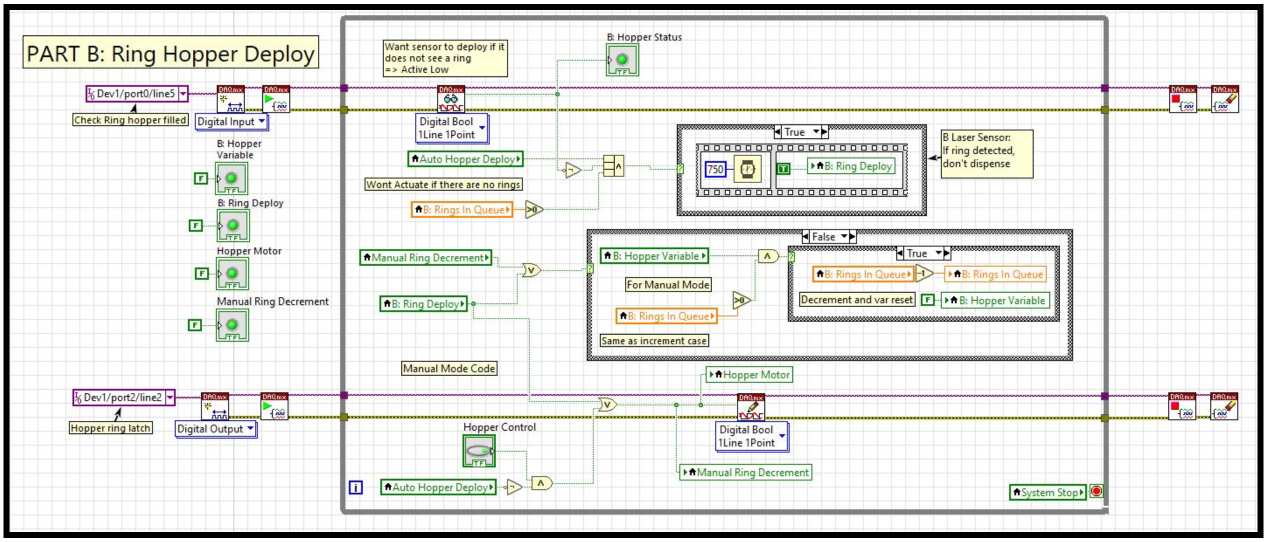

Figure 3.3

Parts C and D are combined since the pushing found in D depends on what C reads. The two main sensors found in C are the capacitive sensor that reads completes, and the inductive sensor that reads metals. The truth table of the needed logic is found in figure 3.4a. As a baseline case, the solenoid in part D will always push whatever it senses in front of it. It will only deactivate when “C/D: Complete Allow Var” is activated. That variable is high only if both the capacitive and inductive sensor is high. The proximity laser in C acts as a signal to save whatever that capacitive and inductive sensors read. A negative edge detection boolean crossing is used to reset the flag variables at the proximity laser in D.

As mentioned before, the other functions like the counters and manual mode has been implemented in the same way as the other parts. That negative edge detection in part D increments the success cases, and the actuator in D returning to its rest case (negative edge) increments the fail case.

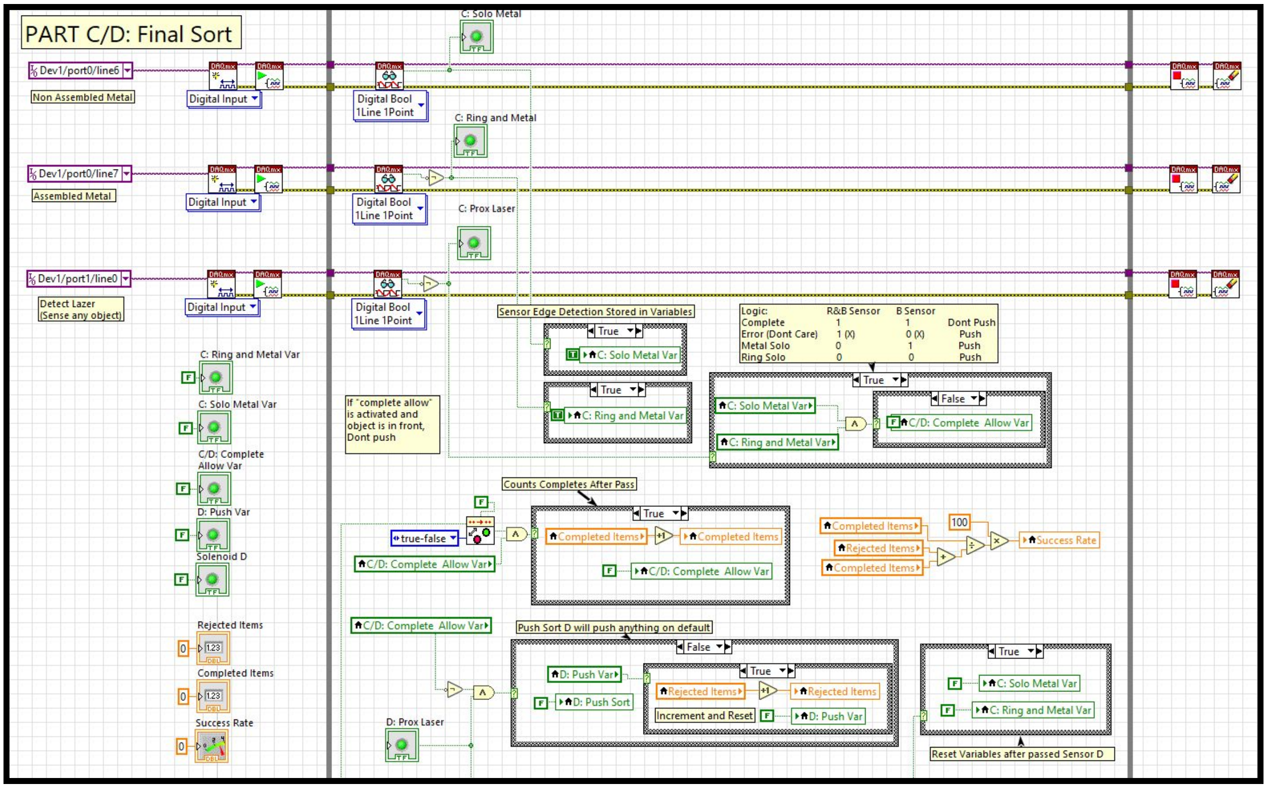

Figure 3.4a

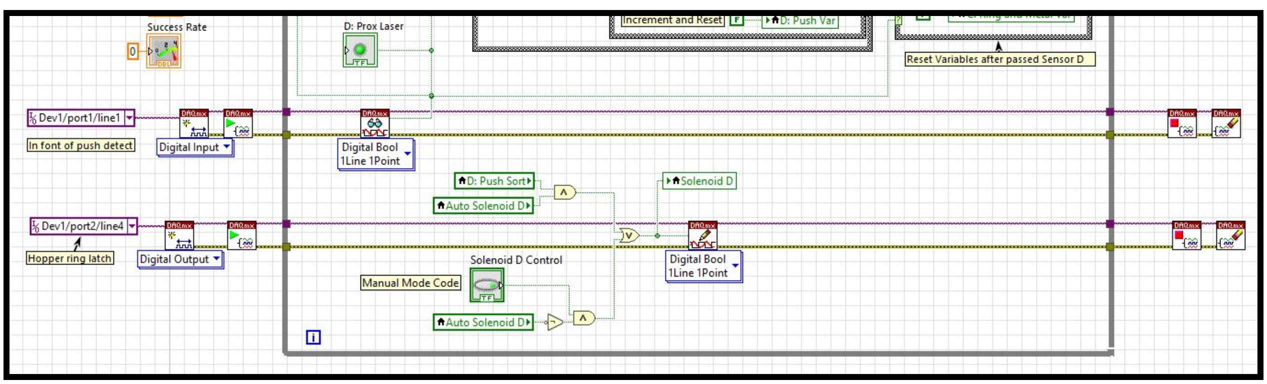

Figure 3.4b

Conclusions

Different types of applications require different types of sensors to collect data from the environment and actuators to provide force to move physical objects. The main focus of this project was to write a code in LABVIEW to control Industrial Control Technology unit (ICT3) through a DAQ device. Performance of ICT is very similar to baggage handling systems. Metal Pegs and Plastic Rings have the role of luggage, Belt Conveyor and Chain Conveyor could represent the baggage carousel in airports. Therefore, this project was completed with high standards and all the safety precautions are included in this report.

References

- http://www.bytronic.net/product/industrial-control-trainer-ict3/
- http://www.ni.com/tutorial/4594/en/