Can you hear me?

OBJECTIVE

After completing this assignment, the student will be able to:

a) Identify and use internal control bit instructions
b) Enter multiple rung programs

A 10
B 11
C 12
D 13
E 14
F 15

REFERENCES

Cox, Technician's Guide to Programmable Controllers
Allen-Bradley, Bulletin 1766 User Manual

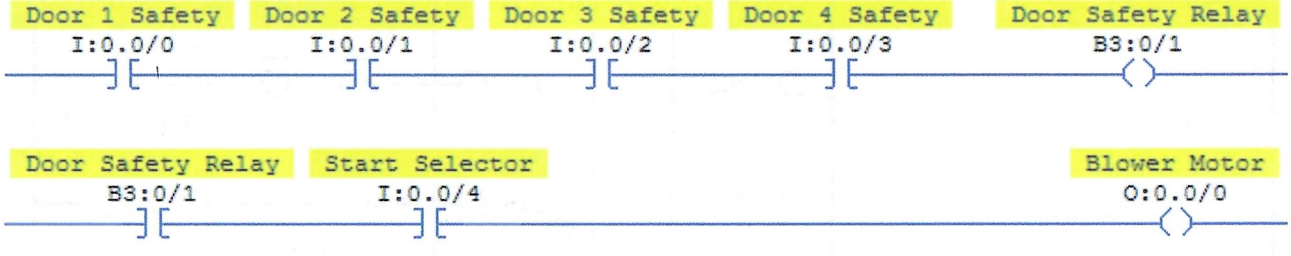
MATERIALS NEEDED

Allen-Bradley MicroLogx 1400 programmable controller with simulator
RSLogix 500 and RSLinx Software

DISCUSSION

When a coil and contact set are needed, but an external output is not required, the internal relays are used. Typical applications would be to store information, for programming convenience, or to overcome program limitations. In the MicroLogix 1400 these instructions are called internal control BITS. They are located in default Data File 3. MicroLogix 1400 Bit addresses start at B3:0 and can be expanded out to B3:255. Bit address can use the full addressing convention, such as B3:1/0 or a short form of address, like B3/16. Both of these addresses refer to the same bit. If you started counting at the first available bit, B3:0/0 or B3/0 bit 16 would be B3:1/0 or B3/16. Remember to start with zero when making your count or the numbers won't line up.

An example of information storage is shown below. If you had a machine that had four doors on it and to be safe you wanted to make sure all of the doors are shut before it is allowed to run, a storage relay could be used. There may be several rungs of logic that need to make sure the doors are closed so instead of writing the logic with the four door switches in series several times it can be done once to a storage relay and then a single XIC with the storage relay address can be used to check the status.

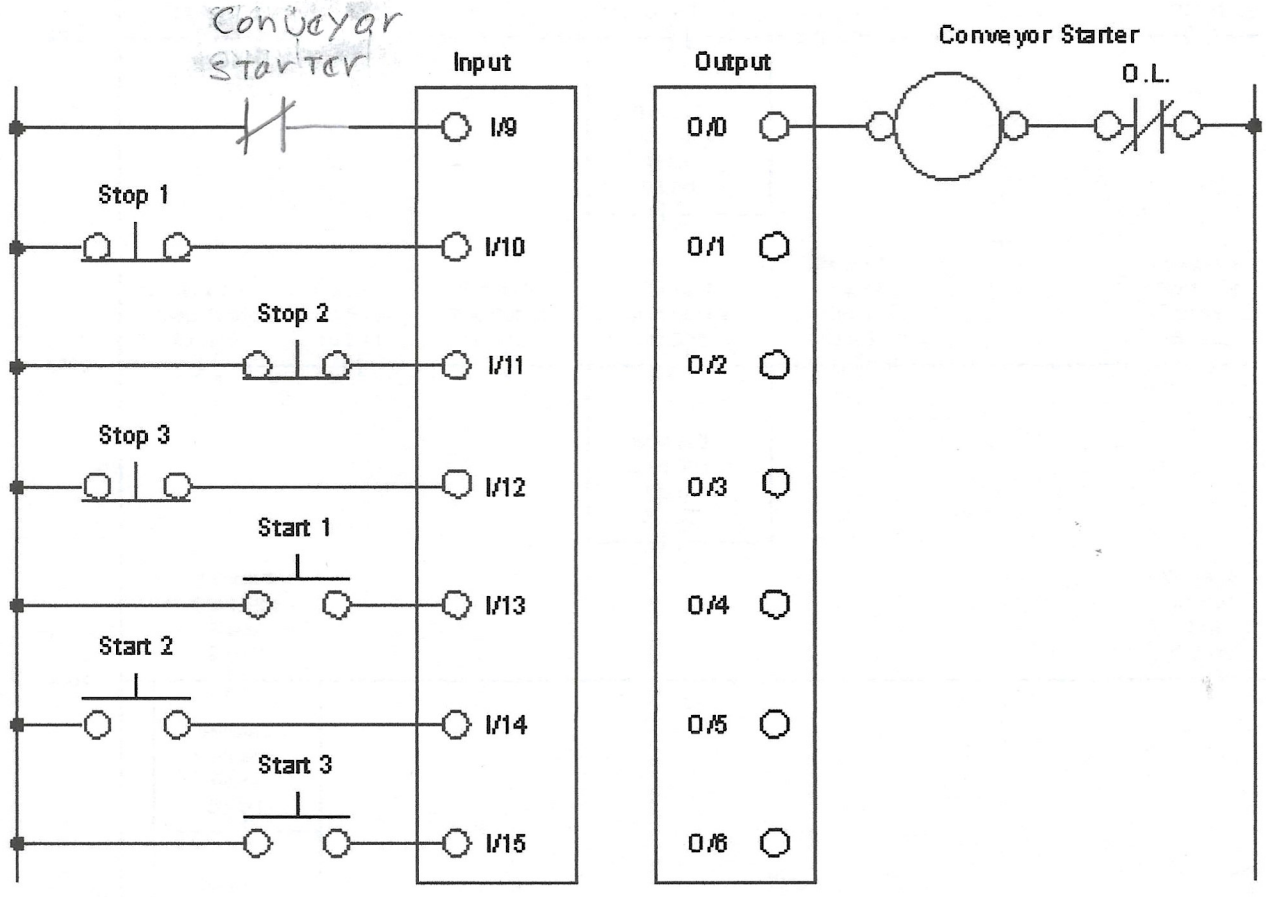


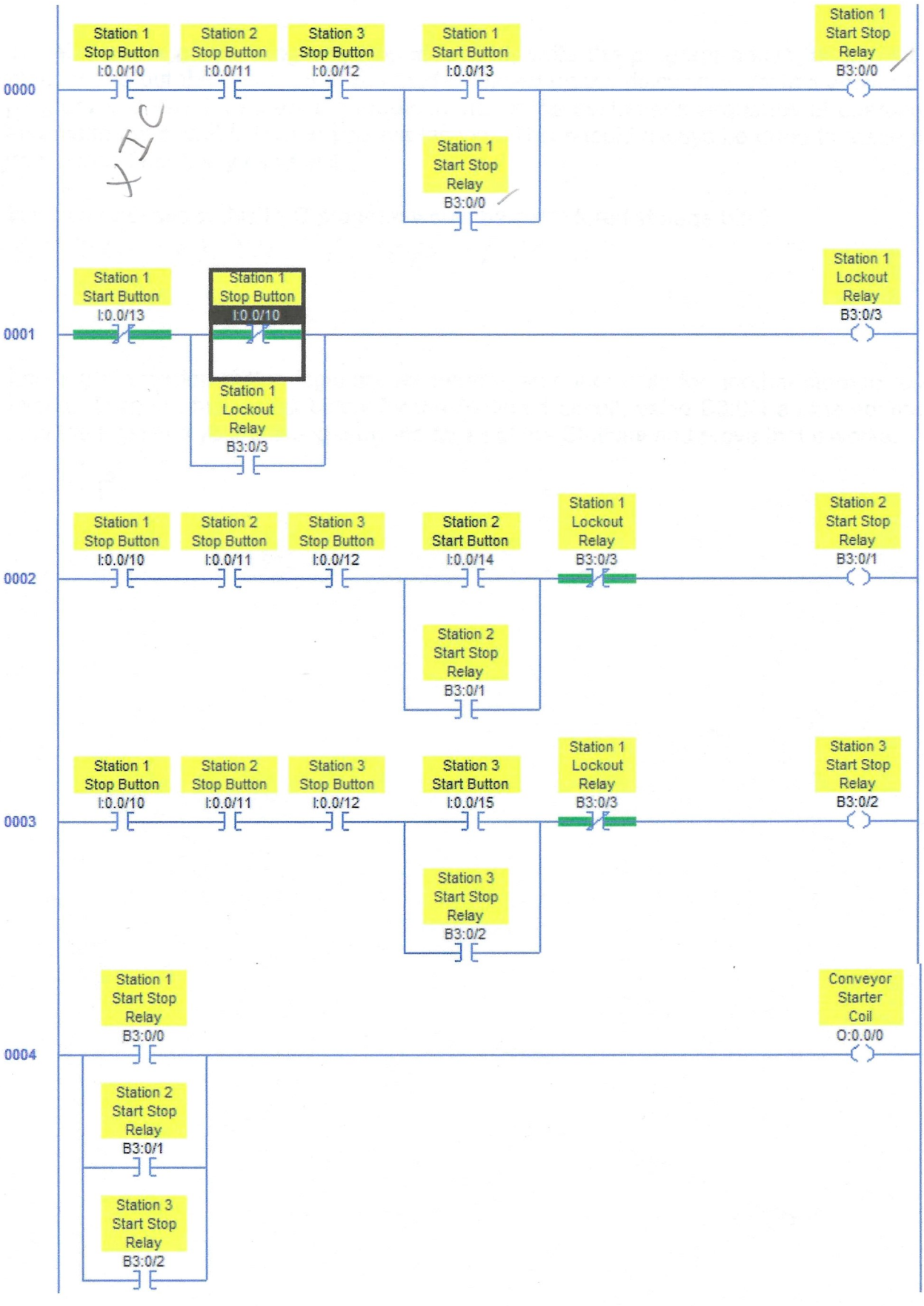
If anyone of the doors opens the door safety relay will be off so any rung that uses a single XIC with it will go false, just like it would if the rung had the four door safety switches in series on that rung.

A Practical Application

A distribution warehouse has a conveyor system that spans two rooms. In room number 2 there are several people working and two 3 wire start stop stations, number 2 and 3, are located there. In room number 1 only one person is working and it has one 3 wire start stop station, which is number 1. In room number 1 the only time the worker there stops the system is if there is a problem and in room number 2 workers frequently start and stop the conveyor. Since they can all see each other, workers can start the system from either station 2 or 3, but they visually check to make sure no one will get hurt if they do so. They have no way of making that same visual check on the poor worker in room number 1, so if the system gets stop from station 1, the system needs to only be started back up from station 1. This will protect the lone worker in room 1.

This is the start of any programming job you'll ever get into. You must understand the sequence of operation. A brief description like the one above is typically what you would get from a customer. As a programmer you must be able to convert this sequence of operation into working PLC code. Below is a program that meets the requirements of the sequence of operation described above that uses only instructions and techniques you've already learned: Refer to the wiring diagram below:




0000
Station 1 Stop Button I:0.0/10
Station 2 Stop Button I:0.0/11
Station 3 Stop Button I:0.0/12
Station 1 Start Button I:0.0/13
Station 1 Start Stop Relay B3:0/0
Station 1 Start Stop Relay B3:0/0
0001
Station 1 Start Button I:0.0/13
Station 1 Stop Button I:0.0/10
Station 1 Lockout Relay B3:0/3
Station 1 Lockout Relay B3:0/3
0002
Station 1 Stop Button I:0.0/10
Station 2 Stop Button I:0.0/11
Station 3 Stop Button I:0.0/12
Station 2 Start Button I:0.0/14
Station 1 Lockout Relay B3:0/3
Station 2 Start Stop Relay B3:0/1
Station 2 Start Stop Relay B3:0/1
0003
Station 1 Stop Button I:0.0/10
Station 2 Stop Button I:0.0/11
Station 3 Stop Button I:0.0/12
Station 3 Start Button I:0.0/15
Station 1 Lockout Relay B3:0/3
Station 3 Start Stop Relay B3:0/2
Station 3 Start Stop Relay B3:0/2
0004
Station 1 Start Stop Relay B3:0/0
Conveyor Starter Coil O:0.0/0
Station 2 Start Stop Relay B3:0/1
Station 3 Start Stop Relay B3:0/2

In RSLogix, create a new program, named LAB05, write the program above, document it, download it and put the controller in run mode. Based on the discussion section above; test the program and see if it meets the requirements of the customer's sequence of operation. Try and make the circuit fail when you are testing. This should always be done to make the program doesn't have any flaws in it.

Q1. Which addresses in this PLC program would be considered storage bits?

B3:0/0, B3:0/1 B3:0/2 B3:0/3

Q2. Looking at the stop button logic above, there is an opportunity for another storage relay circuit. Diagram that circuit below for the Station 1 circuit, using B3:0/4 as the address. Add the logic into your existing program for all of the Stations and prove that it works.

skip

Q3. What storage relay in the program disallows station 2 or 3 to start the system when it was stopped by station 1? Explain the logical process that it uses to do this.

B3:0/3 Station 1 Lock out Relay. The XIO examine The output of B3:0/3 we want it to Read True if output is 0 or open

Q4. The most important thing a programmer needs to consider when writing logic is "what if?" What if the operator pushes buttons in a different sequence? Could that produce a dangerous situation that could damage equipment or injure or even possibly kill someone? This is a big responsibility, not to be taken lightly. In the labs using simulators it is very unlikely that anything could be done to hurt anybody, but we want to watch and see if our program being tested has the potential to do so in a working situation. With this in mind, consider the conveyor starter coil rung of logic. What if that starter was being commanded to run and it tripped out on overload protection? The PLC wouldn't have any way of knowing that had occurred so when a maintenance person presses the reset button on the starter, the conveyor would immediately start back up. This would be an unexpected event and people may have their hands in a spot on the conveyor that could cause injury. What device on the starter could be wired into input 9 that could be used to know that the starter coil had dropped out? Draw that device into the wiring diagram on page 2 and label it. Hint: Every starter will have this on it without purchasing any special accessories or attachments.

the NC aux contact

Q5. Based on your answer from question 4, use that device in your program, wired to input 9, to prevent the system from being started back up unexpectedly. Explain your solution below and download it to the controller, then prove that it works with the simulator.

SKIP

Attach a title page and program printout to this lab before turning it in for a grade.