2017

Research And Development Of Industrial Integrated Robotic Workcell And Robotrun Software For Academic Curriculum

Siddharth Y. Parmar
Michigan Technological University, syparmar@mtu.edu

Recommended Citation

Parmar, Siddharth Y., "Research And Development Of Industrial Integrated Robotic Workcell And Robotrun Software For Academic Curriculum", Open Access Master's Report, Michigan Technological University, 2017.
https://digitalcommons.mtu.edu/etdr/373

RESEARCH AND DEVELOPMENT OF INDUSTRIAL INTEGRATED ROBOTIC WORKCELL AND ROBOTRUN SOFTWARE FOR ACADEMIC CURRICULUM

By

Siddharth Y. Parmar

A REPORT

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

In Mechanical Engineering

MICHIGAN TECHNOLOGICAL UNIVERSITY

2017

This report has been approved in partial fulfillment of the requirements for the Degree of MASTER OF SCIENCE in Mechanical Engineering.

Department of Mechanical Engineering – Engineering Mechanics

Report Co-advisor: *Aleksandr Sergeyev*

Report Co-advisor: *Craig Friedrich*

Committee Member: *Scott Kuhl*

Department Chair: *William W. Predebon*

Table of Contents

List of Figures

List of Tables

Acknowledgement

I would like to express my sincere gratitude to my advisor, Dr. Aleksandr Sergeyev for his continuous support and guidance for my research project. His innovative ideas and constant encouragement to improvise have been my driving force to achieve the project goals.

I would also like to thank Dr. Scott Kuhl, Mr. Joshua Hooker and Mr. Vincent Druschke for their earnest collaboration in the “RobotRun” software’s development. I thank Dr. Craig Friedrich for his support and guidance. I have been fortunate to work with Mr. Nick Hendrickson who manufactured parts for the robotic workcell. I am thankful to my fellow undergraduate student Mr. Jacob Staniszewski for his help during the workcell assembly and lab assistance. The timeous tech support from FANUC has been very important throughout my research.

Finally, I would like to thank my family and friends without whom, this could not have been possible.

Abstract

Robotic automation is consuming the laborious tasks performed by workers all over industry. The increasing demand for trained robotic engineers to implement and maintain industrial robots has led to the development of various courses in academia. Michigan Tech is a FANUC Authorized Certified Education Training Center for industrial robot training. This report discusses the research and development of an integrated robotic workcell consisting of three Fanuc robots, Allen Bradley programmable logic controller (PLC), Mini-Mover belt conveyor and Fanuc iR-vision system. The workcell allows students to explore an environment similar to industry and intended to be used for laboratory hands-on activities in two robotic courses: Real-time Robotic Systems and Industrial Robotic Vision System. To complement hands-on activities and to meet the need of educating robotics to those without access to physical robots, an open source robotic simulation software "RobotRun" has been created in collaboration with a faculty member and students from Computer Science department. The features and a few training examples on the software have also been presented.

1. Introduction

Industrial Automation is currently making a huge impact on the global economy. There has been a tremendous amount of growth in the worldwide sales of industrial robots. A recent article on the website of the International Federation of Robotics apprises that a new record sale of 248,000 units was set in 2015 which had an increment of 12% compared to 2014 [1]. Statistics claim that by 2018, 2.3 million units will be installed in factories around the globe [1]. With the rapid growth in the industrial automation sector, there is high demand for trained engineers with up-to-date knowledge in the field of robotics. Adam Stienecker stated, “Today, industry is much less in need of robot designers and much more in need of experts in the application of robots and the design of the systems that work with the robots such as end-of-arm-tooling and vision systems” [2]. He rightly highlights that industry needs more system designers who have the knowledge to interface multiple robots with vision systems, experience with PLC and are aware of different hardware used alongside robots.

Ruminating about the impact of industry growing towards robotic automation, Dr. Sergeyev et.al has received a three-year NSF ATE award DUE-1501335, in the amount $702,324, titled “University, Community College and Industry Partnership: Revamping Robotics Education to Meet 21st Century Workforce Needs” [3]. The project aims to help meet the nation’s forthcoming need for highly trained Industrial Robotics workers. Strategies include developing, testing, and disseminating an updated model curriculum, laboratory resources, and simulation software package suitable for use in both 2- and 4-year Electrical Engineering Technology (EET) programs. The detailed objective of this project highlights updating the EET curriculum to include skills in industrial robotics relevant to current industry needs, enhancing the existing Industrial Robotics laboratory at Michigan Tech to demonstrate the value of hands-on training experiences and developing a free of cost “RobotRun” software for adaptation by other institutions [4].

Michigan Technological University offers two certified robotics courses: Real-Time Robotic systems (EET 4144) and Industrial Vision robotic systems (EET 4147). They provide hands-on training to students on robots and vision systems. The ongoing curriculum and implementation of vision has been presented in detail by Dr. Aleksandr Sergeyev who developed the certified courses for industry representatives and academic curricula for university students [5]. As a part of the robotics vision course, Siddharth Parmar has described a few lab exercises he developed for hands-on training and proposed further developments of the robotics laboratory [5].

Imparting education to students using highly equipped laboratories and providing them the confidence to tackle different applications or troubleshooting systems has been the driving force in developing an integrated robotic workcell. The main objective is to give the students a closer view and a real time experience of the industrial applications.

Many educational institutions have no access to expensive robotic equipment to implement such certified courses in robotics and automation. To render such institutions with an opportunity for robotic education, a robot simulation software called “RobotRun” has been created in collaboration with computer science department. With the introduction of the software, students can run robot simulations for understanding robot operation and functions. The software also allows teaching topics of industrial robots to students from high schools and community colleges. The work discussed here meets the requirement of an updated curriculum, laboratory activities and software simulation package for the NSF grant.

2. Existing Robotic Workcells

Companies such as ABB, FANUC America and KUKA Robotics have designed educational robotic carts for hands on learning of certificate based robotic courses. High schools and universities collaborate with such robot manufacturers to setup an industrial automation laboratory at their institutions. Most of these robots are incorporated in a single cart with a restricted work envelope that limits the robot from being used for a variety of applications. To overcome these limitations, educational institutions build robotic workcells consisting of various sensors, conveyers, controllers and variety of robots. For example, Adam Stienecker developed a laboratory at Ohio Northern University by procuring individual robots from KUKA Robotics and setting up an integrated system with a CNC machine, conveyor and sensors [2]. Dr. Arif Sirinterlikci's team at Robert Morris University developed a vision based sorting laboratory, consisting of the FANUC vision system, a bowl feeder, linear actuator and proximity sensors [5]. The workcell was primarily created as a learning module for the robotics and automation course (ENGR 4700). William Ferry and Andrew Otieno have designed and developed a low cost bottle capping automation system consisting of PLC, vision system and multiple DENSO robots with the purpose of teaching automation integration of different hardware at Northern Illinois University [6].

3. Integrated Robotic Workcell Description

The industrial automation laboratory at Michigan Tech has four FANUC training carts, each comprising of a FANUC LR Mate 200iC robot, R-30iA Mate Controller, Sony XC-56 camera, air compressor and a computer. These robots have an option for interchangeable end-effectors, such as suction cups and 2-finger parallel grippers, which provide flexibility in developing a variety of applications for the laboratory exercises. The integration of three FANUC robots with a belt conveyor, programmable logic controller (PLC), safety guards, through-beam sensors and vision systems in a single robotic workcell is outlined in this report [8].

For safety of the workcell, the robot speeds have been restricted to 250 mm/sec in the teach mode and collision guard is implemented. Collision guard features stops the robot immediately if a collision is detected. The virtual envelope created around the robot restricts the reach of the robot and prevents it from hitting the safety guards. Figure 1 shows the final design of the robotic workcell. The design of the workcell required keeping the robots in close proximity and the workcell had to be within room dimensions. Since conveyors are widely used in the industry, integrating a belt conveyor in the workcell's design became an obvious choice as it can convey different objects and be used for different applications.

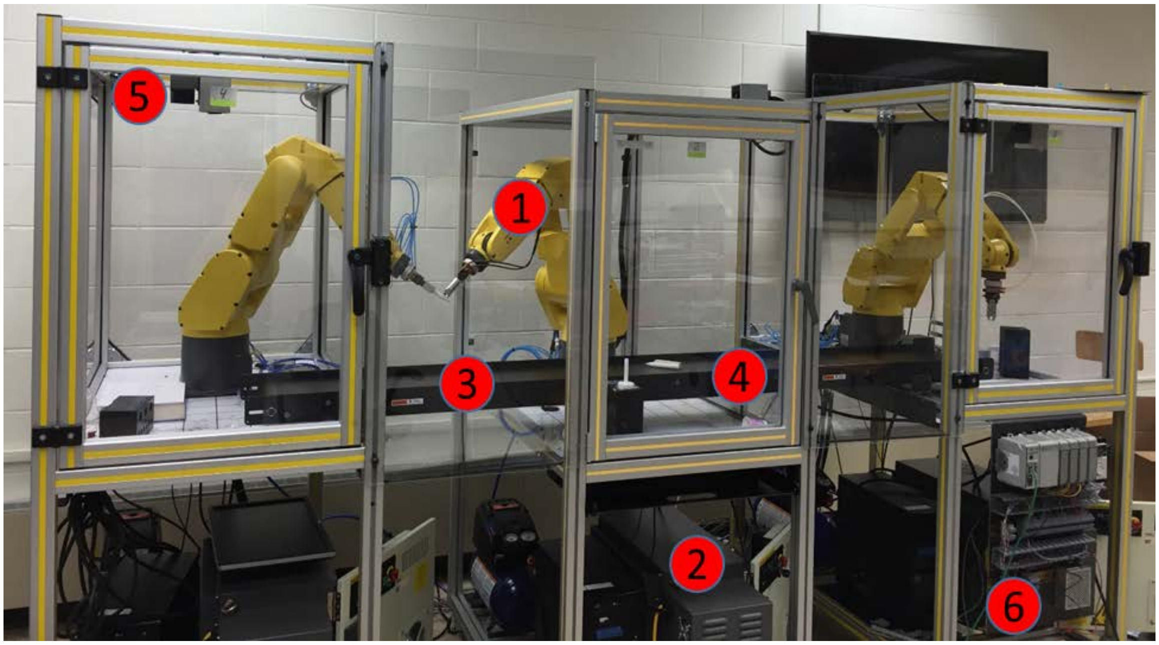


Figure 1. The workcell consisting of 1) Robots 2) Robot Controllers 3) Conveyor 4) Sensors 5) Vision systems 6) Control Panel

3.1 Conveyor System

The conveyor system design was selected based on the various functionalities that would be required to develop industrial scenarios for the lab exercises of the robotic vision course. The black belt was selected for better vision recognition and maintenance reasons. The system conveys numerous products such as Jenga blocks, markers, empty cups and pills. The conveyor can be run either at four different speeds in the forward direction, or at one constant speed in forward and reverse directions. The variable frequency drive (VFD) mounted on the control panel provides an option for setting up multiple speeds. The specifications of the conveyor and its parts are shown in detail in Table 1.

Table 1. Detailed specifications of the conveyor

S No.	**Specifications**	**Description**
1	Conveyor	Manufactured by Mini Mover Conveyors, LITE series model Width – 10”, Length – 84”, Speed 3-43 feet/min
2	Conveyor frame	Hard black anodized aluminum frame with integral 0.70" high side guides
3	Belt	Black, PVC (polyvinyl chloride, low friction cover)
5	Geared-Motor	Manufactured by Oriental Motor Co. Ltd., Brushless DC Motor, Model No. BLM460SP-GFV2, Single phase 100-120 V, Output Power - 60 W
6	Variable Frequency Drive	Manufactured by Oriental Motor Co. Ltd., 115/60 VAC input variable speed Digital Display, Model No. BMUD60-A2

3.2 Vision system and sensors

Nearly any robot currently used in industry is equipped with a vision system. Vision systems are being used increasingly with robotics and automation to perform common and sometimes complex industrial tasks, such as: part identification, part location, part orientation, part inspection and tracking. The vision system provides the robot “eyes” needed to perform complex manufacturing tasks. Extensive usage of vision systems in the automation industry helps achieve high accuracy and speeds for various operations in manufacturing and assembly lines. The Fanuc vision system currently used at Michigan Tech consists of a camera, 2D iRVision package and a spot light. The camera and light used in the workcell are shown in Figure 2 and their specifications in Table 2.

Table 2. Specifications of the vision system

S No.	**Parts**	**Description**
1	Camera	Model No. - XC-56, mfg. by Sony, Black & white CCD camera, 659 X 494 pixel array running at 30 frames/sec, VGA resolution
2	Lens	Model No. – DF6HA – 1B, mfg. by Fujifilm, Focal length 6mm
3	Spot Light	Model No. - LEDWS50L20-XQ High intensity LED White spot light, 3 LED’s

Figure 2. 1) XC-56 Sony Camera 2) LED spot light. These are connected to the robot’s controller.

Sensors are an integral part of most automation systems that detect objects and give feedback to the system’s controller. A through-beam photoelectric sensor has been installed on the conveyor, as shown in Figure 3. It consists of an emitter (emits IR light) and a receiver.

The sensor detects an object when the emitted beam is obstructed and, therefore not captured at the receiver's end. The sensor is used to detect the presence of marker in one of the laboratory exercises discussed in the report. The sensor is mounted on the steel brackets and can be adjusted to suit the height of different objects. The sensor's specifications are presented in Table 3.

Figure 3. Through beam sensors mounted on brackets

Table 3. Specifications of sensor

S No.	**Parts**	**Description**
1	Photoelectric switch receiver	Model No. SSR-0P-4A
2	Photoelectric switch emitter	Model No. SSE-00-4A

3.3 Control Panel and PLC Setup

The control panel, shown in Figure 4, consists of an Allen Bradley PLC (Model No. 1769 CompactLogix L32E) with one input (Model No.1769-IQ16) and 3 output (Model No.1769-OB16) modules, conveyor VFD and Omron SMPS (Switch mode power supply, Output - 24V, 1.1 A). The PLC controls the conditional and sequential operation of the entire workcell in production mode, and interacts with sensors, conveyor system and the robot controllers. It acts as the master controller of the system by sending digital I/O signals to the robot controllers to initialize program execution.

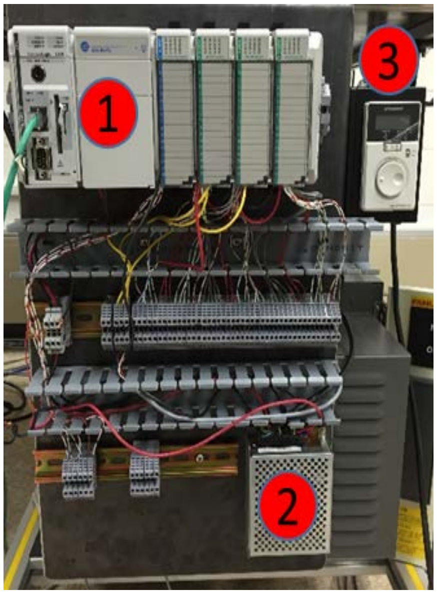


Figure 4. Control Panel's wiring and its components 1) Programmable Logic Controller, 1769-L32E 2) Switch Mode Power Supply, S82J-0224 3) Variable Frequency Drive, BMUD60-A2

The PLC is connected to a computer with Ethernet cable using the TCP/IP protocol and the PLC programming is done on RSLOGIX5000 software (Rockwell Automation) installed on the computer. The panel is mounted on the cart of the FANUC robot and is enclosed safely with Plexiglas guarding. The PLC is powered by the SMPS and is assigned an IP address for communication using Ethernet. The diagram in Figure 5 presents the connections between all modules.

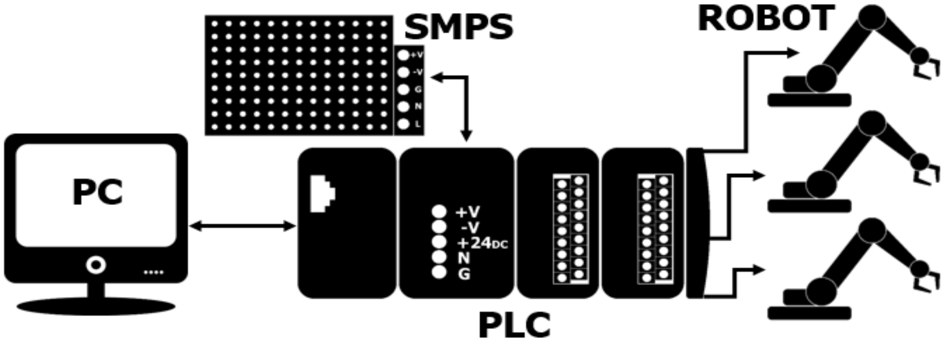


Figure 5. Connection and configuration representation of the following: PC, PLC (Programmable Logic Controller), SMPS (Switch Mode Power Supply) and Robots.

Using the digital I/O (input/output) method of communication, the user can send signals from the PLC to run a program on the robot controller. The PLC digital output modules send on/off signals that are received by the input module of the robot controller. To achieve this functionality, the following are steps involved:

- Configuring and wiring the devices
- Mapping I/O on controller to the connections
- Sending the signal from PLC using ladder logic program

3.3.1 Configuring and wiring the controller devices

The VFD of the conveyor motor is connected to the PLC using source logic, and is used to start/stop and run the conveyor in reverse/forward direction. Three outputs from PLC are connected to the three terminals X0, X1 and X2 on VFD. The significance of each terminal of the VFD used for connections with the PLC is shown in Table 4.

Table 4. Details of the VFD terminals connected to PLC with description of its functionality

Pin No.	**VFD Terminal**	**Function**	**Description**
9	C0	In-COM0	Input Signal common (0 V external power supply)
8	X0	FWD	The motor rotates in forward direction when this signal is turned on
7	X1	REV	The motor rotates in reverse direction when this signal is turned on
6	X2	M0	The two speeds can be selected using this signal
5	C1	IN-COM1	Input Signal common (0 V external power supply)

The connection conversion board is installed on the robot controller and allows communication between any peripheral device and the main board CRMA15 and CRMA16 I/O ports, as shown in Figure 6.

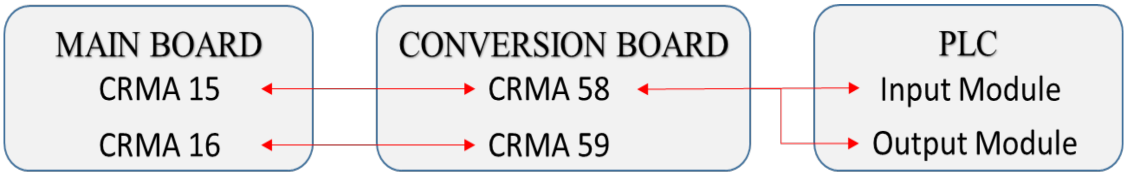


Figure 6. Wiring connections between PLC and robot controller.

The PLC acts as a peripheral device to the robot controller, and the connections between the PLC and the controller are made using the 50 pin Honda Tsushin Kogyo MR-50RFD connector, as shown in Figure 7.

Each digital I/O of the PLC module is connected to the CRMA58 I/O port on the controller using the Honda pin connector. The pin numbers represent the physical digital inputs and outputs of the robot's I/O module. The FANUC controller uses a predefined I/O section called User Operator Panel I/O.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
	19	20	21	22	23	24	25	26	27	28	29	30	31	32			
33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50

Where, Pin no. 1-16 represent DI101-DI116 (Digital inputs)

Pin no. 17-20, 29-30 represent 0 V (Ground)

Pin no. 22-25 represent DI117-DI120 (Digital inputs)

Figure 7. 50 pin Honda Tsushin Kogyo MR-50RFD connector with detailed pin assignments

These UI (user input) and UO (user output) signals are used to communicate with the PLC. The functions of these I/O's has already been configured by FANUC [7] and assigned to dedicated UI and UO numbers, as shown in Table 5. Each of the PLC I/O's is mapped to the digital I/O's of the robot controller, as shown in Figure 8.

Table 5. UOP inputs and outputs to individual commands

UOP Input Signals	**Process I/O UOP UI (User Input)**	**UOP Output Signals**	**Process I/O UOP UO (User Input)**
IMSTP	UI 1	CMDENBL	UO 1
HOLD	UI 2	SYSRDY	UO 2
SFSPD	UI 3	PROGRUN	UO 3
FAULT RESET	UI 5	PAUSED	UO 4
HOME	UI 7	HELD	UO 5
ENBL	UI 8	FAULT	UO 6
PNS1	UI 9	ATPERCH	UO 7
PNS2	UI 10	TPENBL	UO 8
PNS3	UI 11	BATALM	UO 9
PNS4	UI 12	BUSY	UO 10
PNS5	UI 13	SNO1	UO 11
PNS6	UI 14	SNO2	UO 12
PNS7	UI 15	SNO3	UO 13
PNS8	UI 16	SNO4	UO 14
PNSTROBE	UI 17	SNACK	UO 19
PROD_START	UI 18	RESERVED	UO 20

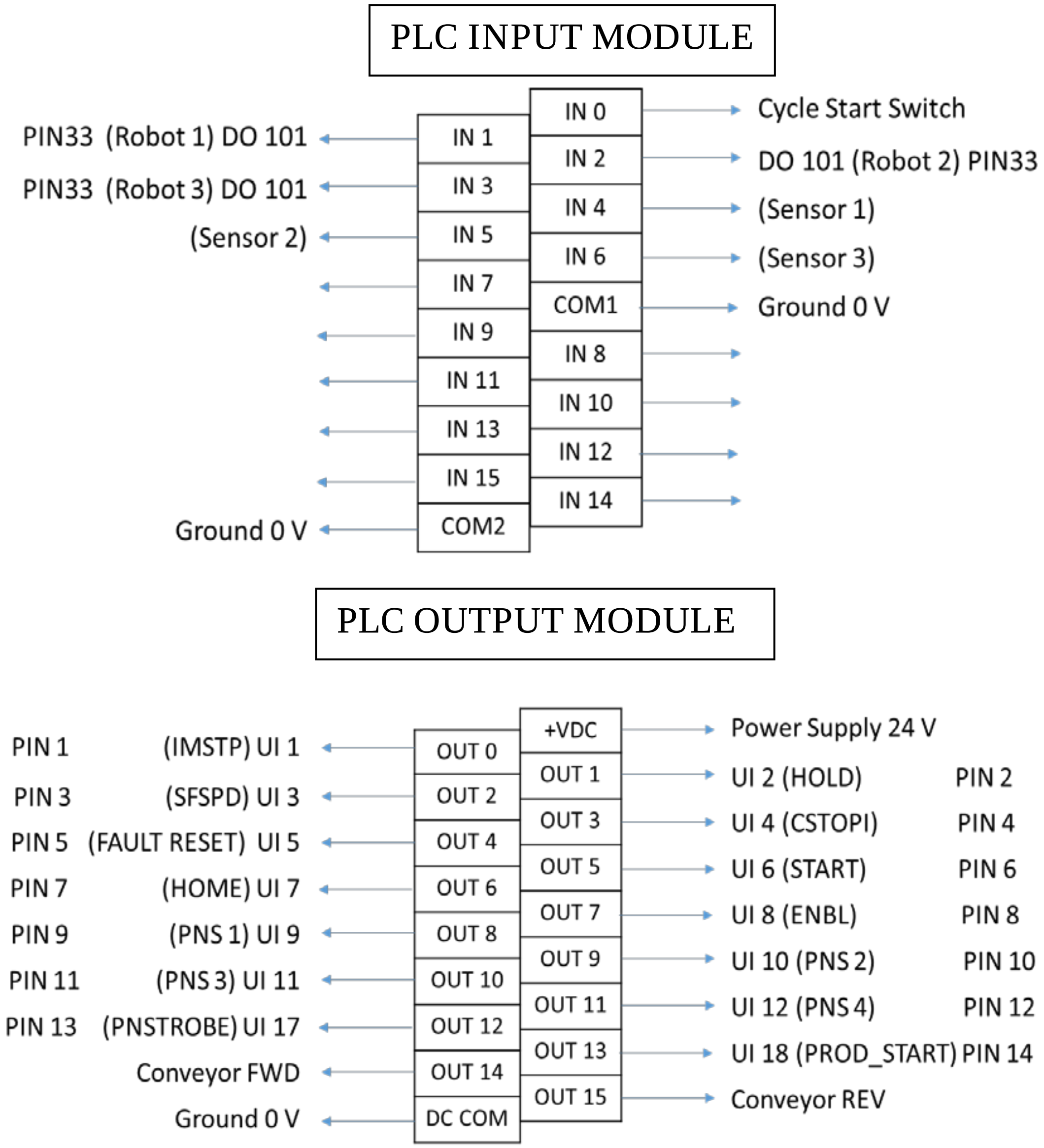


Figure 8. PLC I/O to UI/UO connections

3.3.2 Mapping I/O on controller to the connections

Once the physical connections between PLC and robot controller is established, the detailed pin assignment needs to be achieved by mapping the individual I/O's to the pins in the controller. The following steps on the teach pendant are performed for mapping I/O's:

- Press MENU. Select I/O and select the TYPE, UOP.
- Press F4 for switching between Inputs and Outputs.
- Press F3, CONFIGURE to see a screen similar to Table 6.

➢ In the Range column, the UI range is entered.

➢ Rack and Slot refer to the position of the I/O module on the controller.

➢ Start refers to the Pin no. of the 50-pin Honda connector that is being assigned to the respective UI or UO.

➢ Status indicates the current state of the I/O and will display any of these three options: Assigned, Unassigned and Pending.

➢ If there is any I/O with the status pending, then restarting the controller will automatically set the status to assigned.

Table 6. Details of the UI's assigned

#	RANGE	RACK	SLOT	START	STATUS
1	UI [1— 1]	48	1	1	ASSIGNED
2	UI [2— 2]	48	1	2	ASSIGNED
3	UI [3— 3]	48	1	3	ASSIGNED
4	UI [4— 4]	48	1	4	ASSIGNED
5	UI [5— 5]	48	1	5	ASSIGNED
6	UI [6— 6]	48	1	6	ASSIGNED
7	UI [7— 7]	48	1	7	ASSIGNED
8	UI [8— 8]	48	1	8	ASSIGNED
9	UI [9— 12]	48	1	9	ASSIGNED
10	UI [17— 17]	48	1	13	ASSIGNED
11	UI [18— 18]	48	1	14	ASSIGNED
12	UI [13— 16]	0	0	0	UNASSIGNED

➢ After mapping the UOP I/O’s, the system controller is configured so that the signals are sent from the PLC to control the operation of the robot.

➢ To configure it, Press MENU.

- Select SYSTEM and select the TYPE, CONFIG.
- A list of options is displayed and change the following as shown.

1) Enable UI signals: TRUE
2) START for CONTINUE only: TRUE
3) CSTOPI for ABORT: TRUE
4) Abort all programs by CSTOPI: TRUE
5) PROD_START depend on PNSTROBE: TRUE
6) Detect FAULT_RESET signal: FALL
7) Remote/Local Setup: Remote

3.3.3 Sending the signal from PLC to controller

The PLC ladder logic program, shown in Figure 9, initializes the basic signals that are required to run the robot program.

- At the start, Immediate stop (IMSTP), Cycle stop input (CSTOPI) and Safety speed signals (SFSPD) are turned on and activated for production.
- The fault is reset and the robot is enabled.
- The program on the robot is saved with the name PNSxxxx and the signals from sent from PLC are read as binary inputs by the robot controller. For example, to execute program PNS0011 the signals sent by PLC are PNS1 (2^0), PNS2 (2^1) and PNS4 (2^3).
- After the program number is read by the controller, the ‘‘Program number select strobe signal’’ (PNSTROBE) selects the program on the robot. “Production start signal” executes the program in production mode.

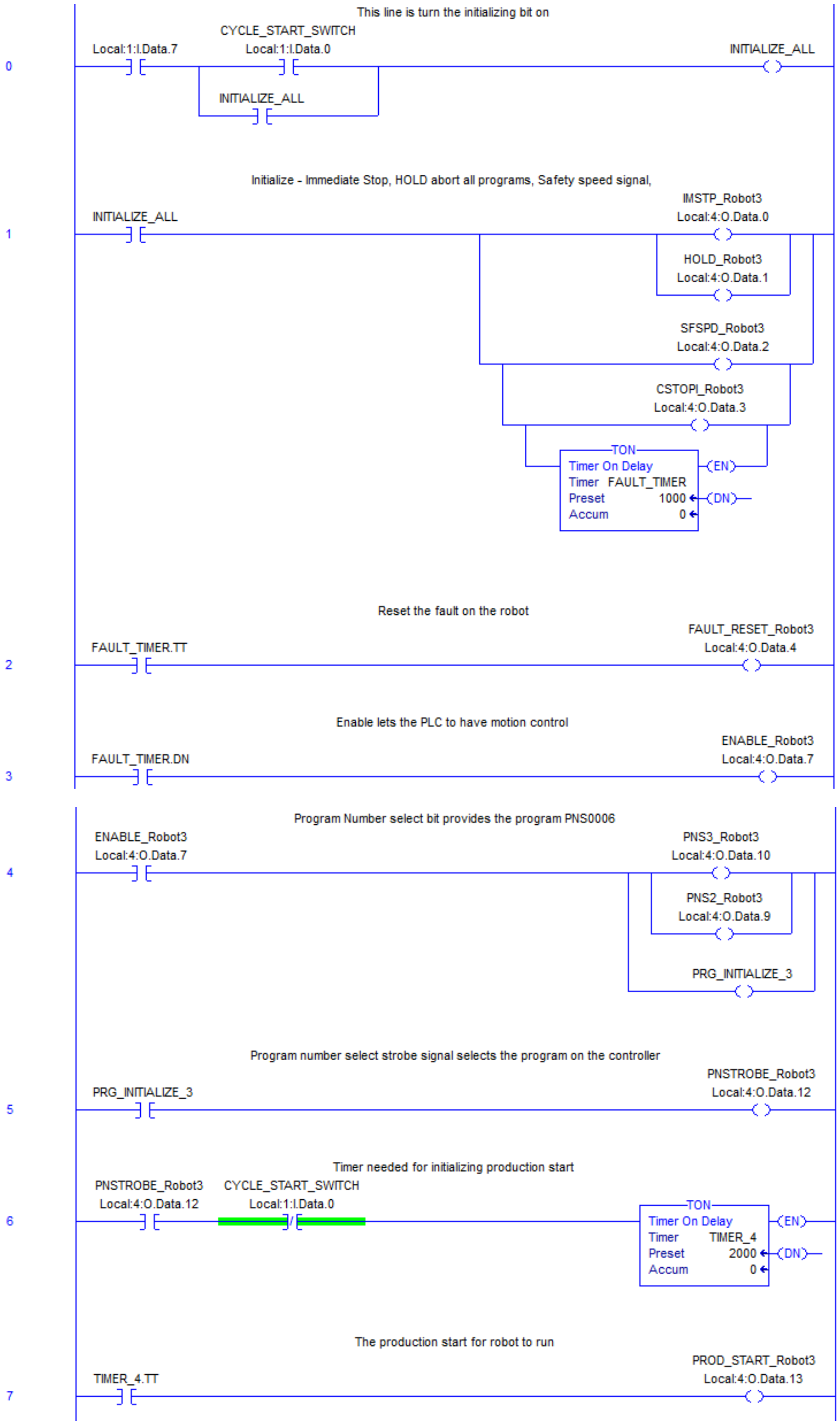


Figure 9. PLC Ladder logic to execute PNS0006 on the Robot 3.

4. Application Scenarios of Lab Exercises

With the integrated robotic workcell, a number of applications can be developed to perform tasks such as packaging, manufacturing and assembly of parts. Students can create innovative projects for both the courses. To provide hands on experience to students and explain the integrated system operation, different lab exercises have been developed and implemented as a part of the course. A few exercises which are discussed in the report, implement FANUC iRVision process to identify different objects.

Following are the basic steps of a 2D single view iRVision process:

- Camera setup

Camera setup lets the system know the physical connections made with camera, the way it is mounted and what camera settings to use.

- Camera calibration

Camera calibration is performed using a calibration grid method which provides the location of camera to the robot controller. It establishes a positional relationship with the robot.

- GPM (Geometric pattern matching) tool

The vision process consists of the GPM tool to teach any object’s pattern to the system. The edges of the objects are detected based on the grayscale values and techniques are used to teach the location, orientation and size of the object.

- Robot Programming

A robot program is written using the vision instructions to teach the position of pickup for a certain location and orientation of the object.

4.1 Jenga Blocks Production and Palletizing

This exercise lets students relate to the various palletizing applications that are used throughout the industry. There are a few wooden blocks randomly on the conveyor in any orientation (-180° to 180°) shown in Figure 10.

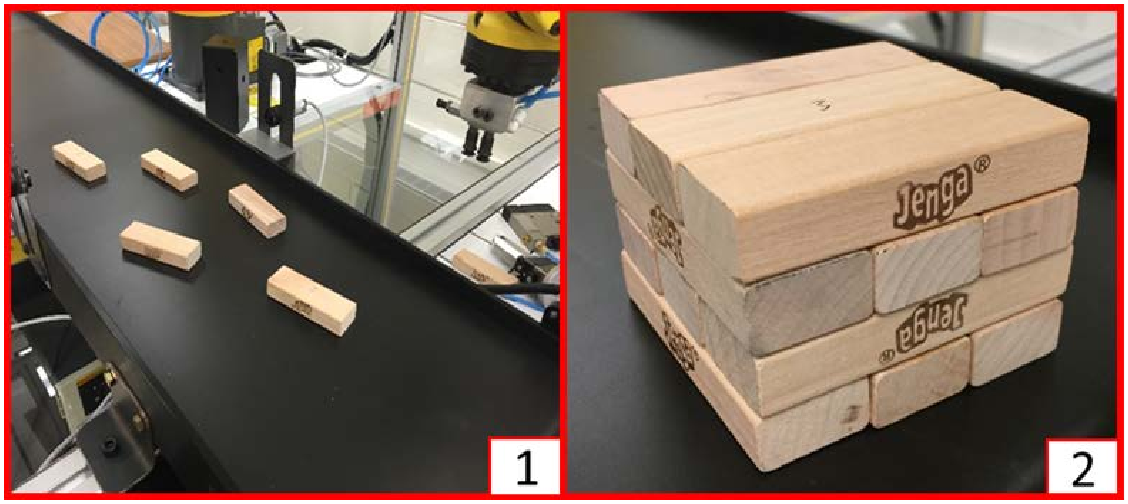


Figure 10. 1) Randomly placed Jenga blocks travelling on conveyor 2) Final Pallet formed by the robot

Initially, the vision system detects the blocks moving on the conveyor. On seeing the first block the controller stops the conveyor. Blocks that are in contact with each other can also be detected. The robotic arm picks up the detected blocks using vacuum cups and forms the final pallet. This is done using the palletizing option provided on the FANUC controller where number of rows, columns and layers of the pallet are defined along with the robot's approach and retreat points from the pallet.

There is a time delay between the camera detecting the block and the conveyor stopping. If within this time the block has moved a certain distance, this error in distance could cause the imperfections in building the pallet. The conveyor speed for the labs is set in the range of 10 - 15 ft/min accounting for the time delay for a good pallet.

The image of the block is taught to the iRVision system's camera mounted above the conveyor and the camera's search window is defined on the conveyor. After the vision process is defined by the students, a robot program is written to integrate the vision with the palletizing program. The final production run is achieved by running the PLC program attached in appendix A, that controls the conveyor and robot motion based on logical instructions.

Having completed this exercise, students learn to optimize robot programming for palletizing applications and PLC programming to run robots in production. It also teaches the iRVision process that involves camera calibration, teaching geometric patterns to the camera and programming the vision instructions.

4.2 Marker pen color sorting and assembly

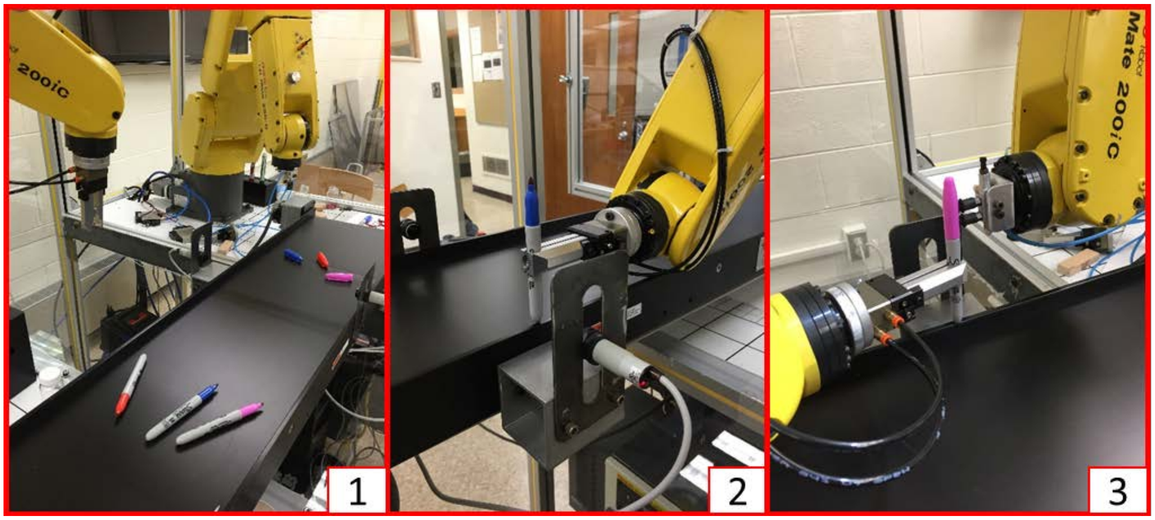


Figure 11. Initial setup with open markers and caps 2) Robot #2 picks the open marker and places it for assembly 3) Robot #3 assembles the cap on the marker

The main objective of this exercise is to teach the system to: differentiate between colors; understand the importance of lighting for the vision system; and safely control multiple robots using PLC. The PLC program used for this application has been attached in appendix A. Three robots are programmed to assemble three different colored markers with their corresponding caps and place them into a packaging tray.

The color of the markers chosen for this scenario are blue, red and pink such that there is enough contrast between the caps to differentiate. The belt being black in color makes it difficult for the robot to detect the dark colors such as blue. The students have to adjust the environment lighting, exposure times and create enough brightness for the camera to detect the blue contrast. The caps are placed in the search region of robot #3 and the open markers are placed in the region of robot #2 as shown in Figure 11.

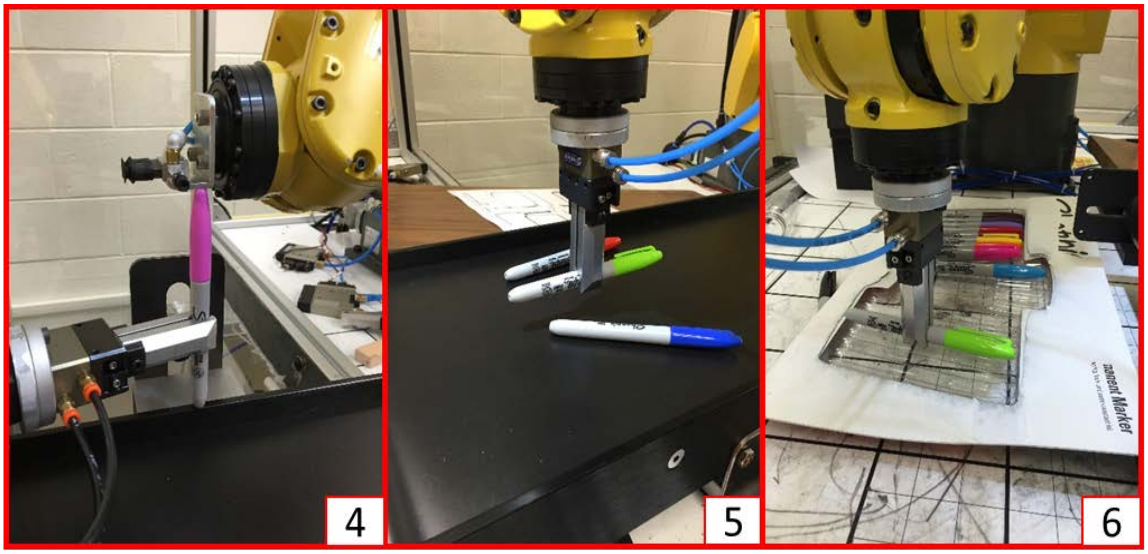


Figure 12. 4) Robot #3 tightens the cap by pressing it on the marker 5) Robot #1 picks the assembled marker 6) Robot #1 places the assembled marker on the tray

The robots' vision system detects position and orientation of the markers and caps in ascending or descending order of contrast (blue, red and pink) based on the detection order settings. Robot #2 picks the marker up and places it on the flat surface for assembly. Through-beam sensor confirms the presence of the marker. Robot #3 picks the cap up from the conveyor, places it on the marker and tightens it on the marker, as shown in Figure 12. After the above process is completed for all three markers, the conveyor starts to move until all markers reach the search region of robot #1. Robot #1 detects the assembled markers and places them on the tray.

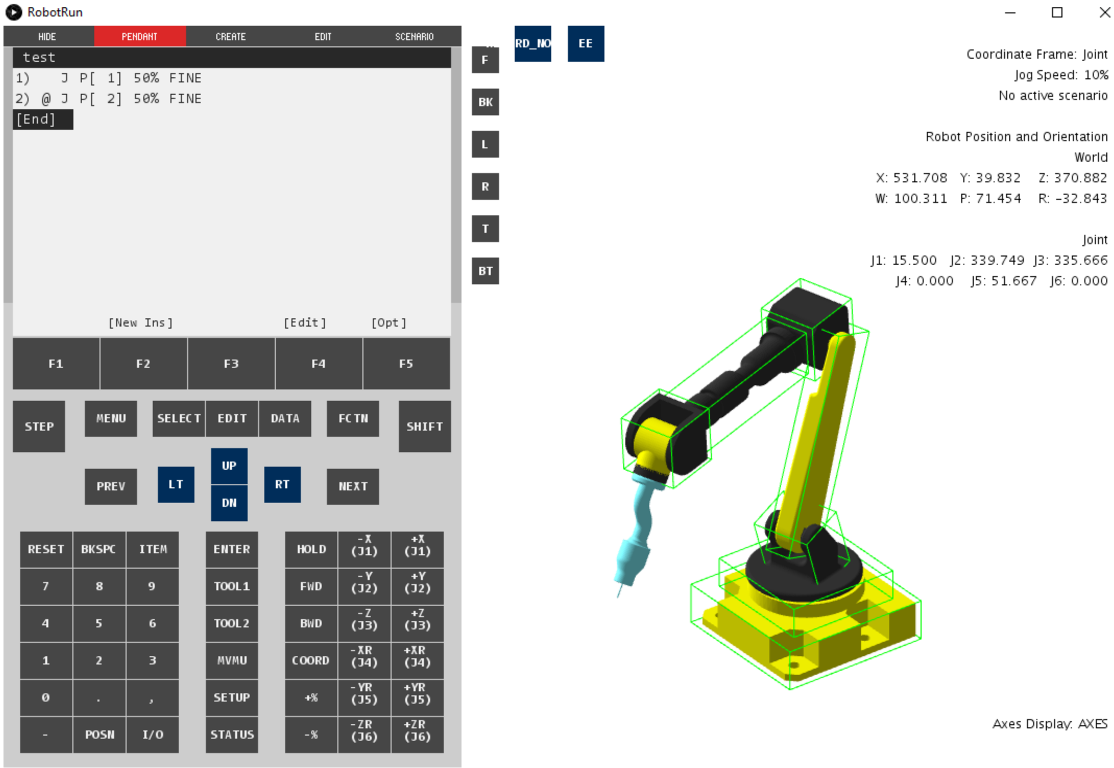
Working on such challenging exercises that involve operation of multiple robots, sensors and vision systems controlled by a master PLC provides an industry-like experience in robot and PLC programming.

5. RobotRun Software

"RobotRun" (http://cs.mtu.edu/~kuhl/robotics/) is an open source free robotic simulation software that is created to enable learning of 6-axis robot programming and operation [10]. This link provides access to the executable MS and Mac file that can be downloaded, data and images used in the software, and the RobotRun Users Guide. It has been developed in close collaboration with a team from the computer science department. The author has contributed by describing the required robot programming platform and it's features to the CS team, worked on testing and troubleshooting the software, and developed the application scenarios.

RobotRun's design and features are inspired from "Roboguide" which is an industrial robotic software provided by FANUC. Roboguide is a very powerful and expensive software with so many features, which are well beyond the scope of an introductory robotic course. For these reasons, "RobotRun" was created with the aim to include all the basic features and functions required for simple robot manipulation and programming. Efforts have been made to provide a similar interface as that of Roboguide software so that students can easily correlate while working on actual robots in the industry.

The software, as shown in Figure 13, displays a 6-axis robotic arm which is controlled by a robotic programming language similar to the actual robot programs. The teach pendant interface used for programming and controlling the robot is highlighted on the left. The screen displays current position, frame and speed settings of the robot on the right. It has been developed on the Java platform and is compatible with Windows and Mac operating systems. All the key features such as frames, registers, position registers, and most of programming instructions taught in the robotics course at Michigan Tech have been implemented in RobotRun.



```
test
1)    J P[ 1] 50% FINE
2)  @ J P[ 2] 50% FINE
[End]
```

Coordinate Frame: Joint
Jog Speed: 10%
No active scenario

Robot Position and Orientation
World
X: 531.708 Y: 39.832 Z: 370.882
W: 100.311 P: 71.454 R: -32.843

Joint
J1: 15.500 J2: 339.749 J3: 335.666
J4: 0.000 J5: 51.667 J6: 0.000

Axes Display: AXES

Figure 13. The interface of RobotRun with a teach pendant module on the left, the 3D 6-axis robot in center and current robot motion settings and position on the right.

Robotic arms generally have a set of tools attached to their face plate. These tools are called end-effectors. End-effectors are used for performing different tasks such as picking up objects, welding or glue dispensing. The software allows to switch between various end-effectors, such as vacuum cup, 2-finger gripper, welding gun and glue dispenser.

The software also lets the user create a simple robotic workcell by importing different representative parts such as tables, metal sheets, cylindrical or rectangular objects from the software library. The user can also add CAD files in stl format to the software's library and import their choice of part in the work environment.

The position and motion of any industrial robot is defined with reference to co-ordinate systems – frames. World, joint, user, tool and jog frames can be configured to optimize robot motion. All these frames can be configured in the software and be used for programming and operation.

5.1 Application Scenarios

The software features discussed above are powerful enough to simulate basic robot functioning required to create applications similar to the industry. Robots are being used across the automation industry for material handling, manufacturing and assembly operations. Efforts have been made to create scenarios that replicate these operations and provide the user a strong foundation of using the different features of the software. Following are the scenarios created using the RobotRun software.

5.1.1 Pick and place objects from multiple stations

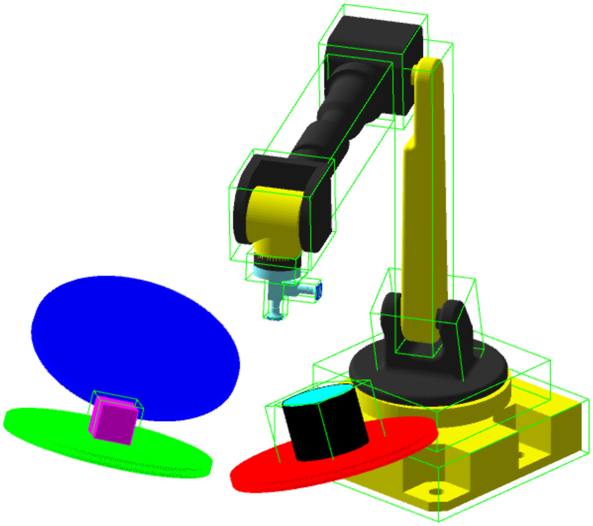


Figure 14. Workcell created to pick and place the box and cylinder on the fixtures in a cyclic manner.

This scenario teaches the user to create a simple robotic workcell using different fixtures and parts, as shown in Figure 15. User has to program and simulate the robot to pick and place parts on fixtures. Two parts are moved between three fixtures in a cyclic manner picking one part at a time using a vacuum cup . The programming involves recording pick and place positions and using I/O instruction to turn the vacuum on and off. The simulation familiarizes user with the environment, teaches basic creation of objects and fixtures and programming robot motion.

5.1.2 Grinding a given part using tool frame and creating a user frame

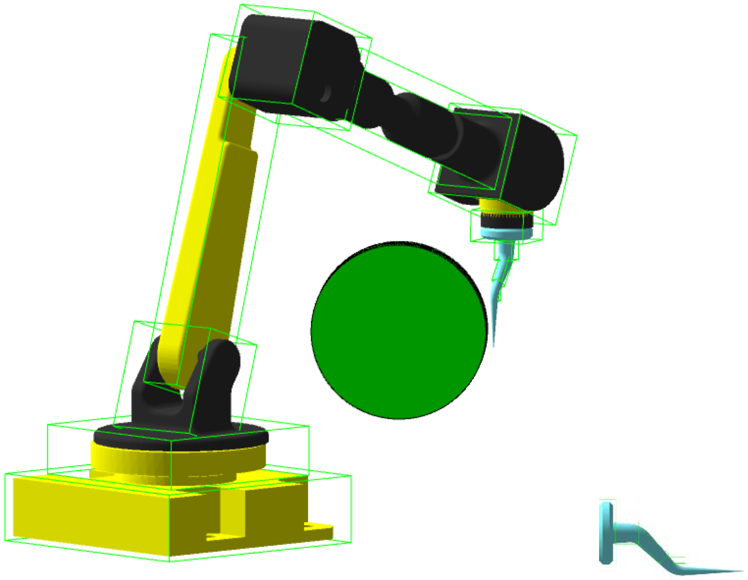


Figure 15. The robot is shown with the part that is grinded on the grinding wheel by rotating using tool frame created.

A crooked shaped part is attached to the robot face plate, as shown in Figure 15, and the conical surface of this part is required to be grinded. While configuring the workcell, the user creates a cylindrical object representing the grinding wheel. The application demands the user to teach a tool frame with the axis of rotation along the pointed tip and use the six-point method. Setting up the tool frame helps the user understand the simplicity and comfort of performing this operation. The user frame creates a separate frame of reference for the robot motion.

The user inserts a rectangular surface in the robot's environment and provides it a random orientation. The task is to create a frame of reference using the edges of this surface. When the user has successfully created the user frame the user can jog the robot along the edges of the rectangular surface. The purpose of these tasks is to teach the importance and application of tool and user frames in a given situation.

5.1.3 Welding application for sheet metal using circular instruction

Welding is an important application where robots are widely used in various industrial processes. This scenario simulates the motion of robot for welding two similar sheet metal parts. The sheet metal part is available in the software library and is imported to the workspace, as shown in Figure 16.

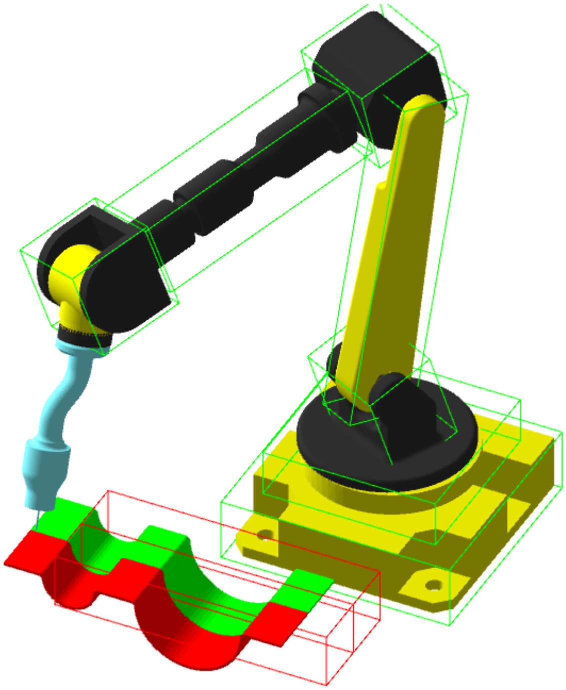


Figure 16. Welding two sheet metal parts along the path requiring linear and circular motion.

The tool used for this operation is a welding tool and user programs the robot to move along the line joining the parts. To accomplish this task, the user first creates the tool frame using three-point method and then uses the circular instruction to program the robot to move in the circular paths. The motion of the welding tool tip is programmed such that it maintains a constant distance from the surface of the sheet metal. With the completion of this scenario, the user has applied the tool and user frames for programming with circular instructions.

5.1.4 Gluing application using position registers and Offset instruction

Gluing is generally performed using the robot by moving in a zigzag motion along the part. The user inserts a rectangular sheet in the workcell and the glue tool is attached for end-of-arm-tooling, as shown in Figure 18. The robot moves along the length of the sheet in zigzag motion, then offsets by a certain value along the breadth and repeats the motion along the length.

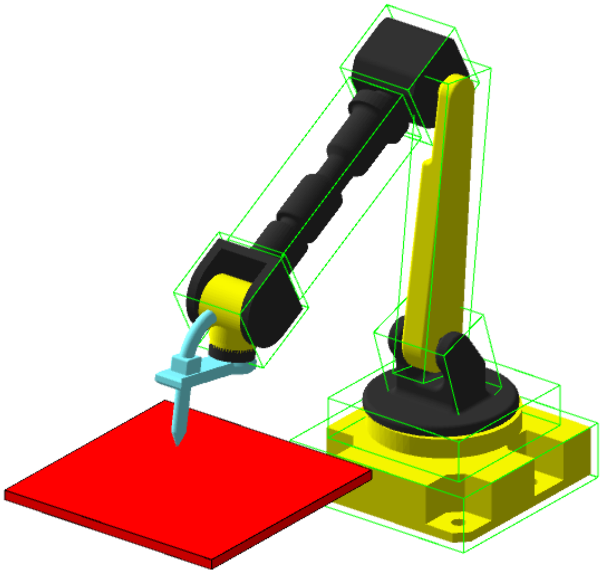


Figure 17. Applying glue on the rectangular sheet in a zigzag manner over the complete area using register and offset instructions.

First, as the robot moves along the rectangular sheet, the user creates a user frame using four-point method. To efficiently program this motion the user implements position registers and records the start position in the program. The offset instruction offsets the value of the position by a certain value and highly simplifies the efforts of programming. The intention of this exercise is to develop efficient programs using some common features of robotic programming such as position registers and offset instructions.

There are few other interesting scenarios that include the usage of copying and pasting feature, macro and register equations. All scenarios have been developed with the purpose of highlighting the features by relating them to real world industrial applications. After the completion of these scenarios, the user would have excelled in implementing basic programming of robots with good understanding of using different features for different applications.

6. Conclusion and Future Work

The report has discussed in detail, a replicable model of an integrated robotic workcell for academia. The curriculum developed based on this workcell is currently being taught to the students at Michigan Tech. Apart from the lab exercises, students are given an opportunity to explore the capabilities of the workcell by doing a course project.

A robotic simulation software – “RobotRun” has been developed to provide a free platform for experiencing industrial robot operation and programming. The RobotRun software is being tested for feedback at faculty and high school student workshops being conducted at Michigan Tech. All the simulation exercises previously performed on “Roboguide” for real-time robotic systems can now be performed on RobotRun.

For future work, the workcell can be upgraded based on various other industrial applications making use of proximity, hall effect and other sensors. Various end effector designs such as multi-head suction cups or multi-finger grippers can be developed as other options. Ethernet adapter option can be implemented for communicating from PLC to robot controllers. 3D area sensors and vision can included in the curriculum for advanced vision techniques such as 3D binpicking. RobotRun will be continuously improvised based on the feedback on user friendliness and suggestions by the users. RobotRun is also being updated with options to handle multiple robots, implement basic vision simulations and developing industrial scenarios using those features.

References

[1] "https://ifr.org/ifr-press-releases/news/world-record," [Online].

[2] A. Stienecker, "Applied Industrial Robotics: A Paradigm Shift.," in *American Society for Engineering Education Annual Conference & exposition*, 2008.

[3] "https://www.nsf.gov/awardsearch/showAward?AWD_ID=1501335," [Online].

[4] A. Sergeyev, N. Alaraje, S. Kuhl, M. Kinney, M. Highum, M. M. and S. Parmar, "Revamping Robotics Education to Meet 21st Century Workforce Needs – Year 2 Progress," in *Proceedings of ASEE*, 2017.

[5] A. Sergeyev, S. Y. Parmar and N. Alaraje, "Teaching Critical Skills in Robotics Automation: iR-Vision 2D Course in Robotic Vision Systems Development and Implementation," in *IAJC/ISAM Joint International Conference*, 2016.

[6] A. Siriinterlikci, A. M. Macek and B. A. Barnes Jr, "Development of a Vision based Sorting Operation Laboratory: A Student Driven Project," in *ASEE*, Seattle, WA, 2015.

[7] A. W. Otieno and C. R. Mirman, "Machine Vision Applications Within a Manufacturing Engineering Technology Program," in *ASEE*, Montreal, Canada, 2010.

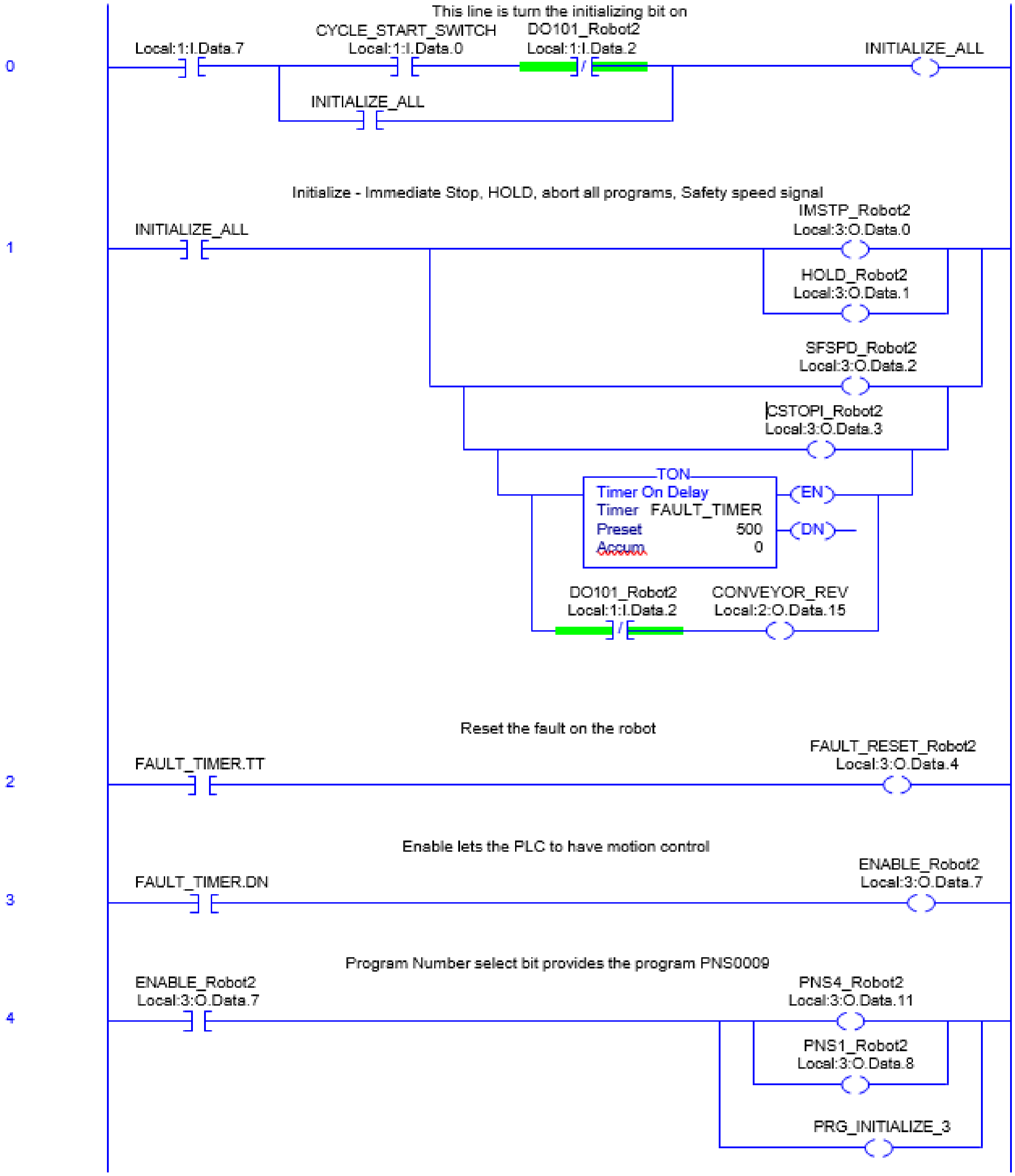
[8] S. Parmar, A. Sergeyev and N. Alaraje, "Teaching Industry Relevant and Application Oriented Skills in Automation and Control by Developing State-of-the-Art Integrated Robotic Workcell," in *Proceedings of ASEE Zone 2 Conference*, 2017.

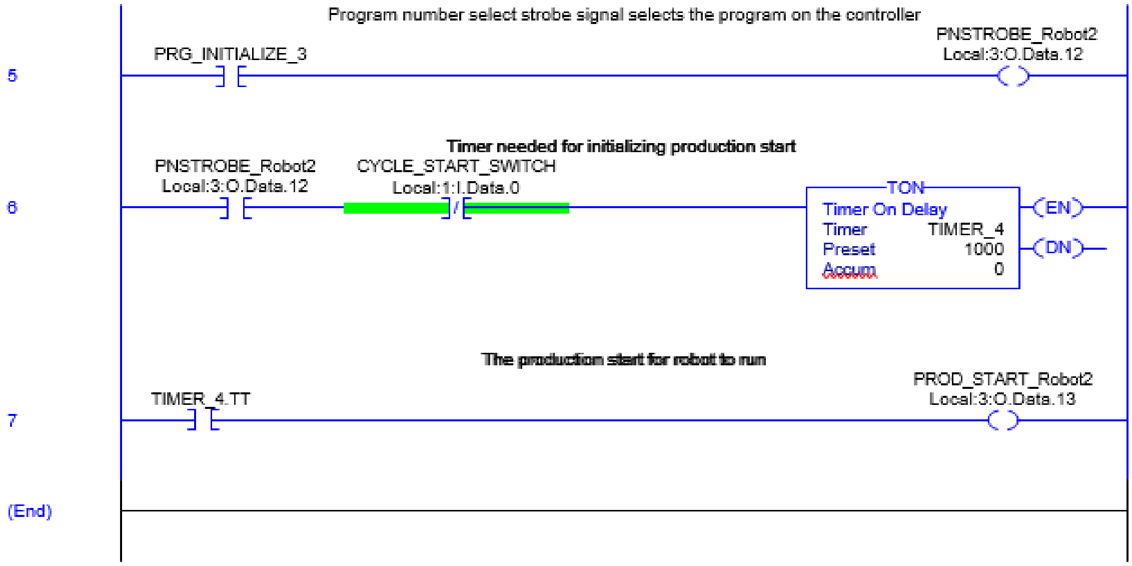
[9] FANUC Robotics System R-30iA Controller HandlingTool Setup and Operations Manual.

[10] A. Sergeyev, S. Parmar, S. Kuhl, V. Druschke, J. Hooker and N. Alaraje, "Promoting Industrial Robotics Education by Curriculum, Robotic Simulation Software, and Advanced Robotic Workcell Development and Implementation," in *IEEE Systems Conference Paper # 1570324750 (Submitted)*, 2017.

Appendix A

PLC program for Jenga lab

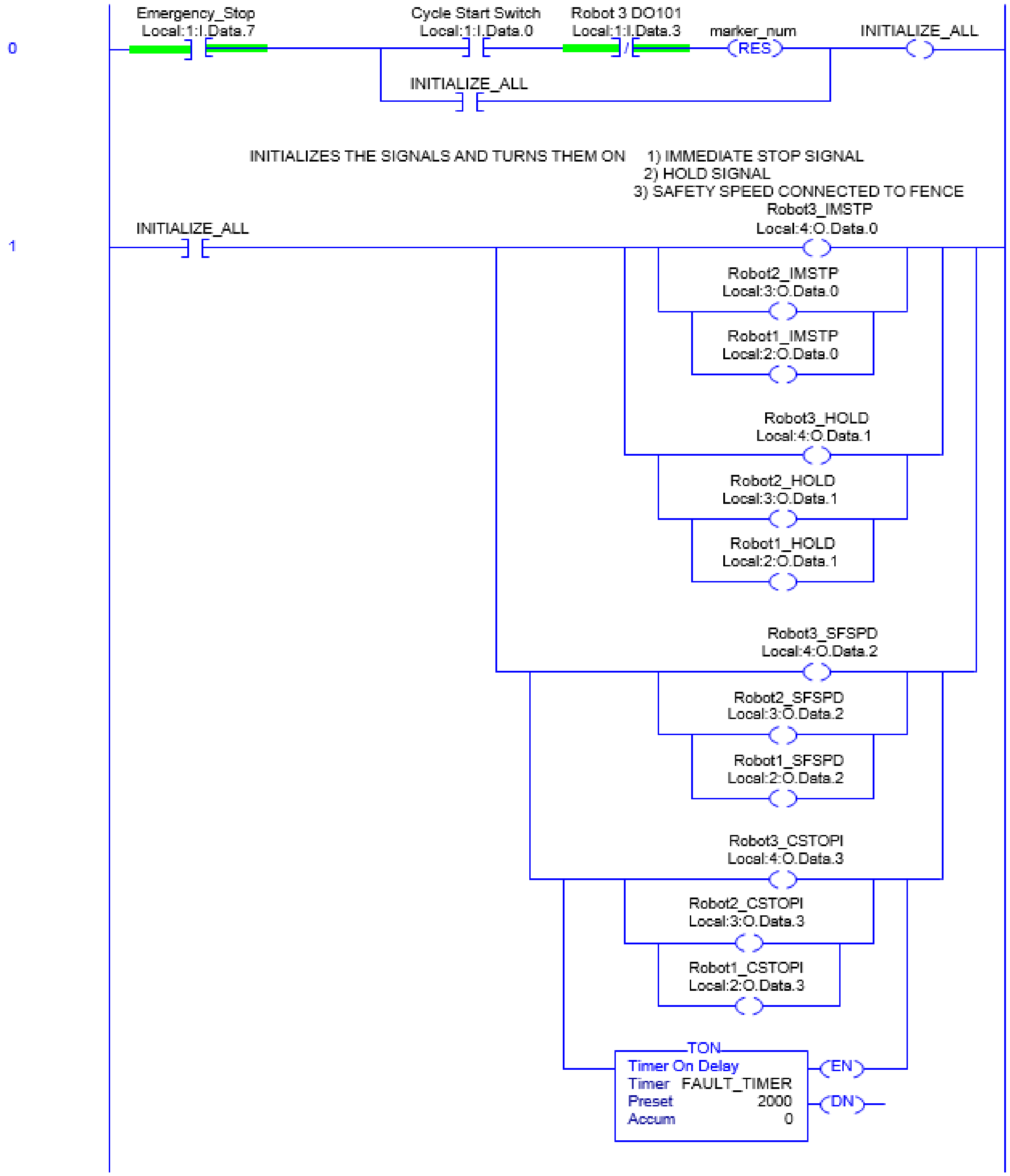


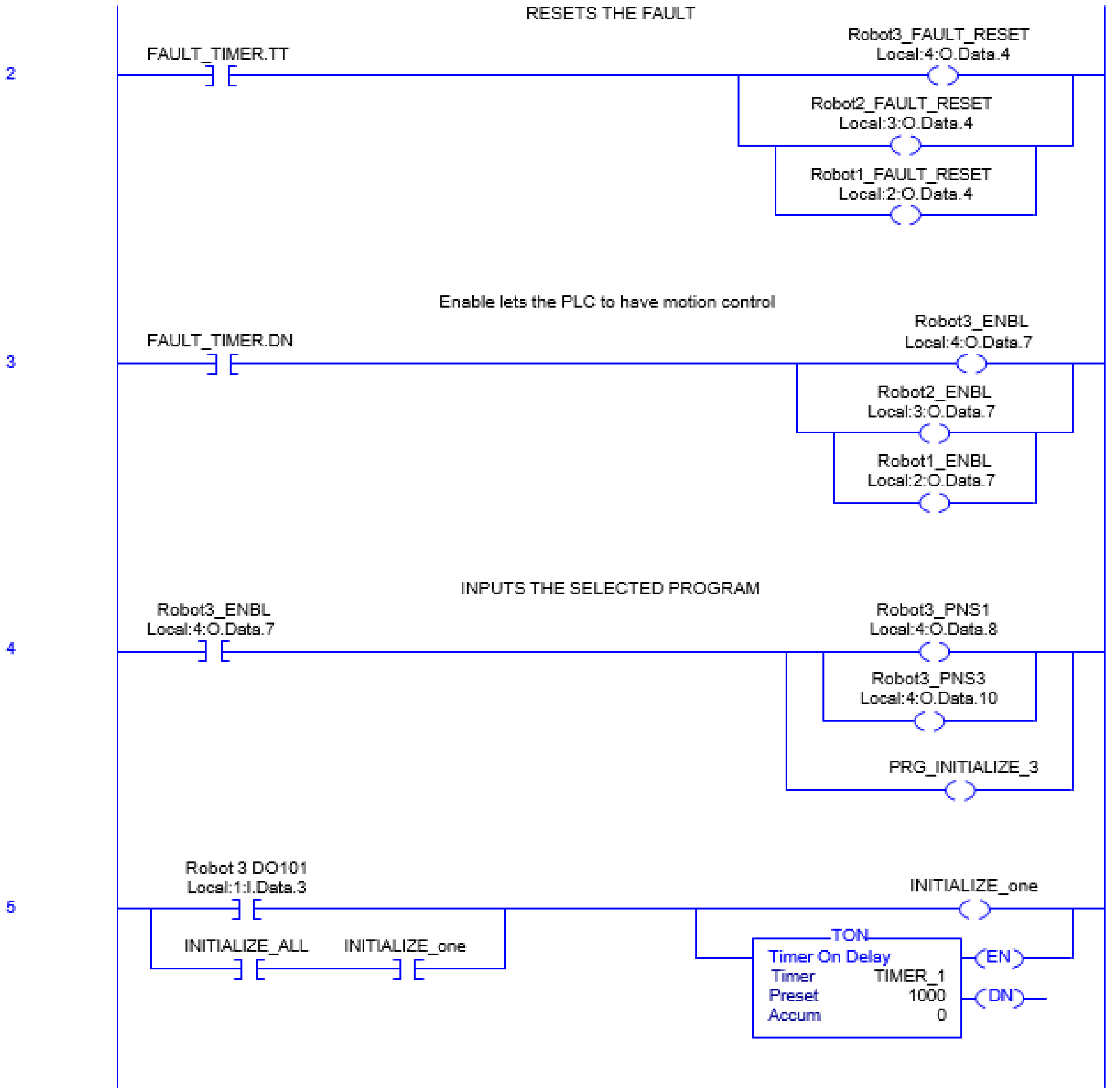

Program number select strobe signal selects the program on the controller

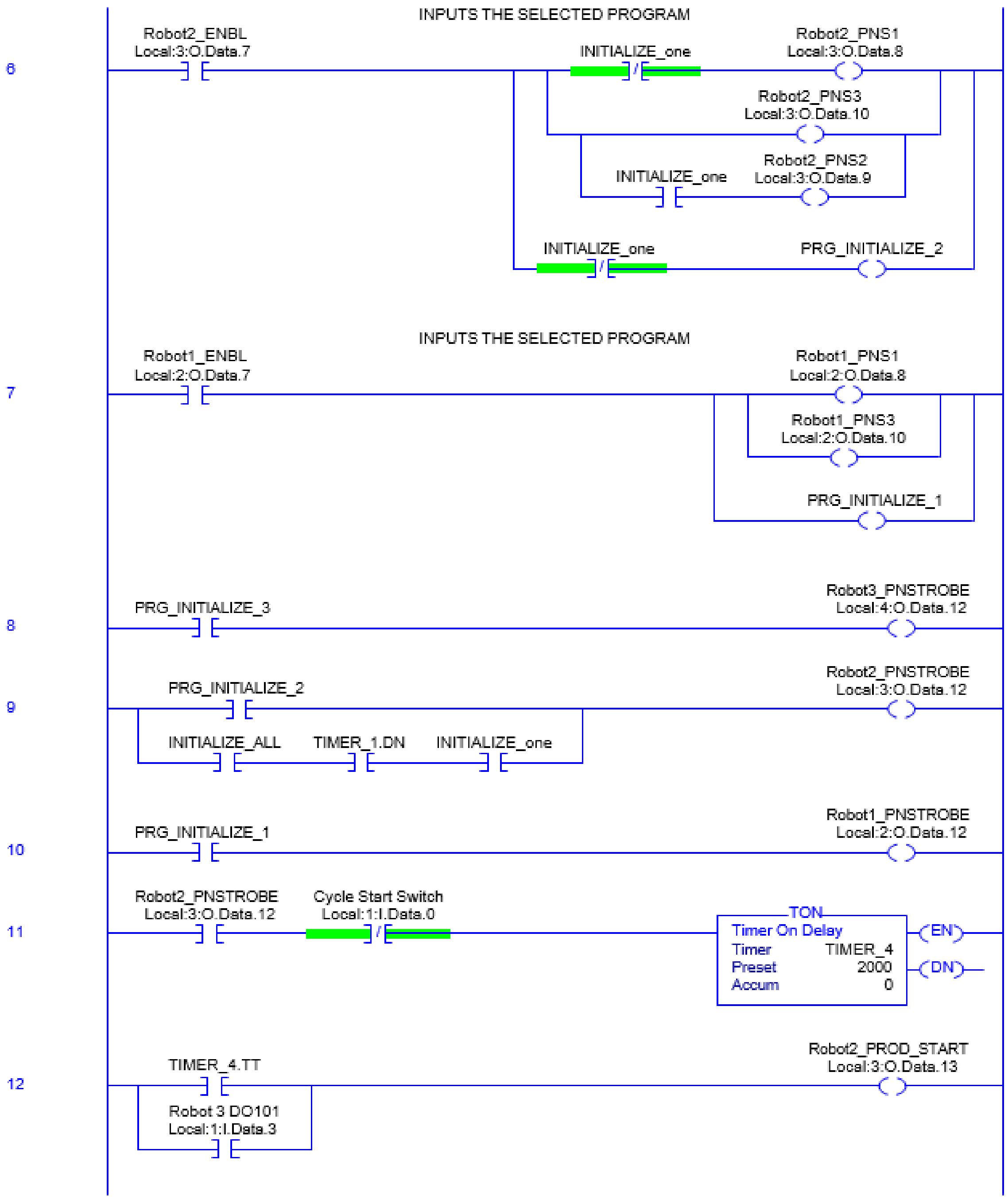
Timer needed for initializing production start

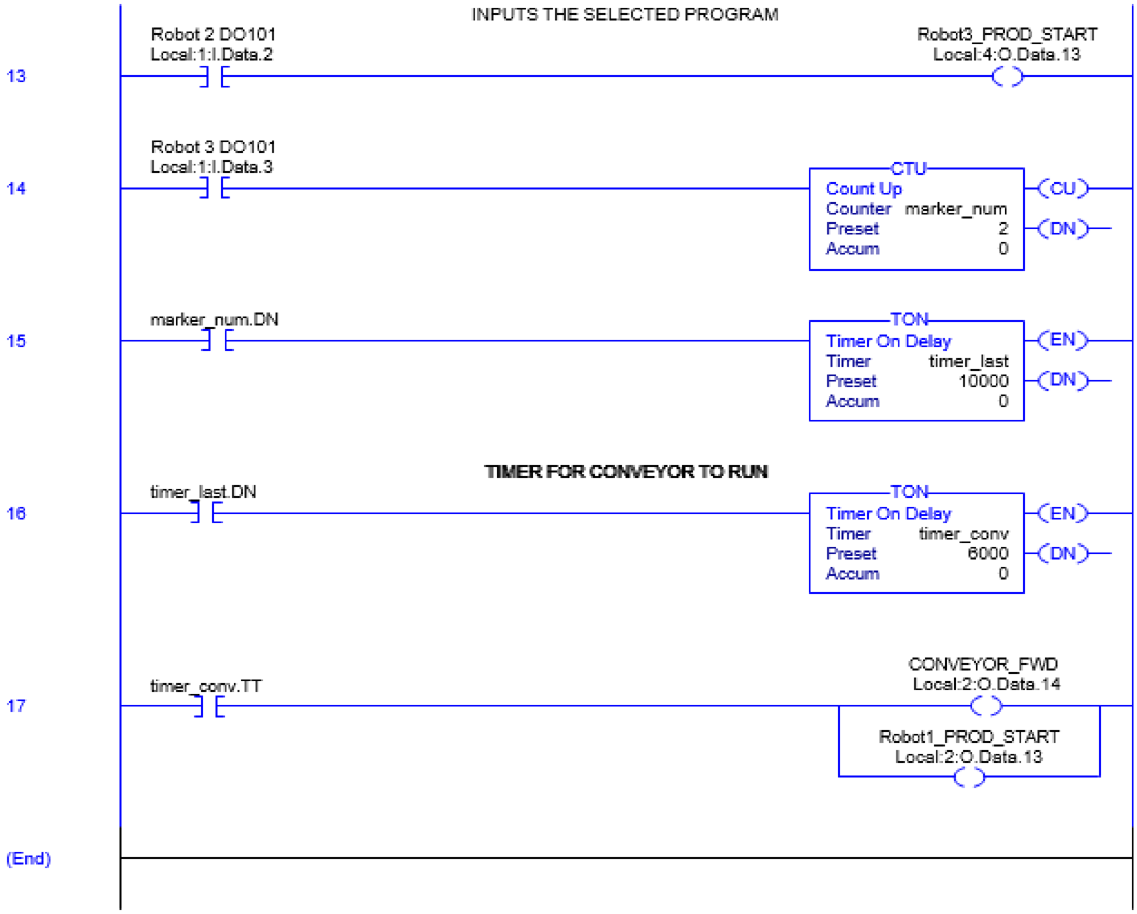
The production start for robot to run

PLC program for Marker Assembly lab




2
RESETS THE FAULT
FAULT_TIMER.TT
Robot3_FAULT_RESET
Local:4:O.Data.4
Robot2_FAULT_RESET
Local:3:O.Data.4
Robot1_FAULT_RESET
Local:2:O.Data.4
3
Enable lets the PLC to have motion control
FAULT_TIMER.DN
Robot3_ENBL
Local:4:O.Data.7
Robot2_ENBL
Local:3:O.Data.7
Robot1_ENBL
Local:2:O.Data.7
4
INPUTS THE SELECTED PROGRAM
Robot3_ENBL
Local:4:O.Data.7
Robot3_PNS1
Local:4:O.Data.8
Robot3_PNS3
Local:4:O.Data.10
PRG_INITIALIZE_3
5
Robot 3 DO101
Local:1:I.Data.3
INITIALIZE_ALL
INITIALIZE_one
INITIALIZE_one
TON
Timer On Delay
Timer TIMER_1
Preset 1000
Accum 0
EN
DN


6
INPUTS THE SELECTED PROGRAM
Robot2_ENBL
Local:3:O.Data.7
INITIALIZE_one
Robot2_PNS1
Local:3:O.Data.8
Robot2_PNS3
Local:3:O.Data.10
INITIALIZE_one
Robot2_PNS2
Local:3:O.Data.9
INITIALIZE_one
PRG_INITIALIZE_2
7
INPUTS THE SELECTED PROGRAM
Robot1_ENBL
Local:2:O.Data.7
Robot1_PNS1
Local:2:O.Data.8
Robot1_PNS3
Local:2:O.Data.10
PRG_INITIALIZE_1
8
PRG_INITIALIZE_3
Robot3_PNSTROBE
Local:4:O.Data.12
9
PRG_INITIALIZE_2
Robot2_PNSTROBE
Local:3:O.Data.12
INITIALIZE_ALL
TIMER_1.DN
INITIALIZE_one
10
PRG_INITIALIZE_1
Robot1_PNSTROBE
Local:2:O.Data.12
11
Robot2_PNSTROBE
Local:3:O.Data.12
Cycle Start Switch
Local:1:I.Data.0
TON
Timer On Delay
Timer TIMER_4
Preset 2000
Accum 0
EN
DN
12
TIMER_4.TT
Robot 3 DO101
Local:1:I.Data.3
Robot2_PROD_START
Local:3:O.Data.13


INPUTS THE SELECTED PROGRAM
13
Robot 2 DO101
Local:1:I.Data.2
Robot3_PROD_START
Local:4:O.Data.13
14
Robot 3 DO101
Local:1:I.Data.3
CTU
Count Up
Counter marker_num
Preset 2
Accum 0
CU
DN
15
marker_num.DN
TON
Timer On Delay
Timer timer_last
Preset 10000
Accum 0
EN
DN
TIMER FOR CONVEYOR TO RUN
16
timer_last.DN
TON
Timer On Delay
Timer timer_conv
Preset 6000
Accum 0
EN
DN
17
timer_conv.TT
CONVEYOR_FWD
Local:2:O.Data.14
Robot1_PROD_START
Local:2:O.Data.13
(End)