The Future of Engineering Education
2024 Annual Conference & Exposition
Oregon Convention Center
Portland, OR . June 23 - 26, 2024
ASEE
Paper ID #42933

Interdisciplinary Senior Design Project to Develop a Teaching Tool: Cobot Integrated Robotic Cell Learning Module

Dr. Yalcin Ertekin, Drexel University

Yalcin Ertekin, Ph.D., CMfgE, CQE is a clinical professor in the College of Engineering, Department of Engineering Leadership and Society at Drexel University, Philadelphia, and serves as the Associate Department Head for Undergraduate Studies for the Engineering Technology program. He received his BS degree from Istanbul Technical University in Turkey, an MSc in Production Management from the University of Istanbul, an MS in Engineering Management, and an MS and Ph.D. in Mechanical Engineering from the University of Missouri Rolla. Dr. Ertekin has also been a Certified Manufacturing Engineer (CMfgE), awarded by the Society of Manufacturing Engineers (SME) since 2001, and a Certified Quality Engineer (CQE) awarded by the American Society for Quality (ASQ) since 2004. In addition to positions in the automotive industry, Dr. Ertekin has held faculty positions at Western Kentucky University and Trine University. In 2010, he joined Drexel University's College of Engineering as an associate clinical professor. He has been instrumental in course development and the assessment and improvement of the Engineering Technology (ET) curriculum, including integrated laboratories, project-based learning, and practicum-based assessment. Dr. Ertekin serves as the faculty advisor for the student chapter of the Society of Manufacturing Engineers (S058). Involved in research, Ertekin has received funding from the National Science Foundation (NSF), private foundations, and industry. His research has focused on the improvement of manufacturing laboratories and curricula and the adoption of process simulation into machining and additive manufacturing practices. His areas of expertise are in CAD/CAM, manufacturing processes, machine and process design with CAE methods, additive and subtractive manufacturing, quality control and lean manufacturing.

Dr. Richard Chiou, Drexel University

Dr. Richard Chiou is Associate Professor within the Engineering Technology Department at Drexel University, Philadelphia, USA. He received his Ph.D. degree in the G.W. Woodruff School of Mechanical Engineering at Georgia Institute of Technology. His educa

Prof. Tzu-liang Bill Tseng, University of Texas at El Paso

Dr. Bill Tseng is a Professor and Chair of the Department of Industrial, Manufacturing, and Systems Engineering (IMSE) at UTEP. He is also a Director of the Research Institute for Manufacturing & Engineering Systems (RIMES), the Texas Manufacturing Assistance Center (TMAC) host institute at UTEP. He received his two MSIE degrees (MFG & DS/OR) from the University of Wisconsin at Madison and a Ph.D. in Industrial Engineering from the University of Iowa. Dr. Tseng is also a Certified Manufacturing Engineer from the Society of Manufacturing Engineers. Dr. Tseng's research area covers artificial intelligence (AI), data analytics, advanced quality engineering technology, additive manufacturing, and systems engineering. Over the years, he has served more than 11 million dollars as a principal investigator sponsored by NSF, DOE, NIST, USDT, DoEd, KSEF, and industries like LMCO, Honeywell, GM, and Tyco Inc. Dr. Tseng delivered research results to many refereed journals such as IEEE Transactions, IIE Transactions, International Journal of Production Research, Journal of Manufacturing Systems, Expert Systems with Applications and other conferences (117 refereed journal articles and 193 refereed conference proceedings). He is currently serving as an editor of the Journal of Computer Standards & Interfaces (CSI), an associate editor of the Asia Pacific Management Review (APMR), and an editor on boards of the International Journal of Data Mining, Modeling and Management (JDMMM) and the American Journal of Industrial and Business Management (AJIBM). He is currently a senior member of the Institute of Industrial Engineers and Society of Manufacturing Engineers and a former chair of the Manufacturing Engineering Division of the American Society of Engineering Education (ASEE). He is also actively involved in several consortia activities.

The desired current set of skills required of modern engineers and technologists has been steadily expanding. In addition to familiarity with production grade industrial robots, collaborative robots (cobots) and automation methods are increasingly becoming essential tools in the design, prototyping and manufacturing of complex systems. In this paper, an inter-disciplinary design project towards the development of a Cobot Integration Learning Module is presented.

The Engineering Technology Department at Drexel University (DU) offers several courses that allow students to interact with robotic and machine vision systems. With many courses focusing on each individual component of production, an opportunity exists to combine several elements to simulate a real-world example of an automated work cell. Students need more skills in programming Cobots and integrating multiple machines into a production process to create innovations in the mechanical, electrical, and industrial fields.

The objective of this project is to develop a small-scale automated work cell that integrates manufacturing machinery into a single integrated learning module. The primary goal of the system is to provide students in the Engineering Technology department with hands-on experience in integrating multiple machines into a single system. The project was inspired by the need to introduce students to small-scale industrialization for artisan companies with budget constraints.

A learning module that was agreed upon was to obtain manufacturing components that were already familiar to students undergoing practical education by machining a Delrin workpiece. This includes a conveyor belt for the transportation of the part, a machine vision camera and photoelectric sensor array for part detection and quality assurance, a 6-degree-offreedom collaborative robot for part transfer, and a CNC for milling operations. The final results demonstrate the possibilities of utilizing readily available commercial products to achieve a desired task.

Furthermore, the work cell's modular design makes it portable, allowing for simple deconstruction and transportation to nearby high schools to generate interest in STEM education among prospective students. Overall, the project aims to provide students with a baseline in how to develop an automated system. The work cell has the potential to be a cost-effective solution for small-scale production with a greater degree of optimization and improvements over time.

Students in the Mechanical, Electrical, and Industrial fields along with many others can learn many new skills from multi-disciplinary projects such as the design and development of a robotic cell. Such projects show students how to use different types of technology and demonstrate how advanced technology such as collaborative robot (cobot), vision camera, cnc machining process, etc. can be integrated in an actual production application. This project teaches future engineers and technologists various advanced skills that can be used in their careers.

Background

In DU's Engineering Technology program, many courses related to robotics, design, and materials are offered to students. Courses such as Robotics and Mechatronics, Quality Control, Manufacturing Materials, Microcontrollers, and Applied Mechanics can benefit from the laboratory experience in applications of mechatronics, robotics, and rapid prototyping. As well as helping in the teaching of various courses, such experience benefits students who are pursuing degrees in the engineering field. Students in the Mechanical-manufacturing, Electrical, and Robotics and Automation concentrations along with many others can learn many new skills from multi-disciplinary projects such as developing a manufacturing cell integrated with collaborative robot and 3-axis cnc machine.

Students in the Engineering Technology programs are required to complete a yearlong three series of capstone course MET 42X Senior Design Project. This three-quarter course sequence aims to train the students in identifying projects of relevance to society, in planning and scheduling a solution, and in entrepreneurial activities that may result from the project. The course is worth three credit hours per quarter offering. The course is also intended to cover an industrial project starting from the proposal writing and conceptual design to final prototype building and concept realization steps. The course is focused on proposal and project progress report writing, prototype fabrication as well as design improvement and optimization. Each quarter, student teams must submit a progress report and demonstrate a physical working prototype at the end of academic year. During fall, winter and spring quarters, they conduct an oral presentation to faculty and practicing engineers from industry. Since this is a capstone project course, many ABET Student Outcomes are assessed each quarter as indicated in Table 1. Written, oral and student contribution rubrics were developed specifically for the capstone project course and are used during assessment and evaluation. The assessor body includes Engineering Technology program faculty, industry advisory board members, sponsoring company engineers as well engineers from various local engineers and invited Drexel University faculty.

Table 1. ABET Students Outcomes assessed per quarter offering.

Engineering Technology Courses		**ABET Student Outcomes**
MET 421	Senior Design Project I	**1, 2, 3, 5**
MET 422	Senior Design Project II	**1, 2, 3, 5**
MET 423	Senior Design Project III	**1, 2, 3, 5**

Project Opportunity and Problem Statement

The Engineering Technology Department at Drexel University offers several curriculums that allow students to interact with robotic and machine vision systems. With many courses focusing on each individual component of production, an opportunity exists to combine several elements to simulate a real-world example of an automated work cell. Students need more skills in programming cobots and integrating multiple machines into a production process in order to create innovations in the mechanical, electrical, and automation fields.

The design team completed research on the growth of the cobot market size. The market size of cobots has been increasing significantly in recent years, and the predicted market size for 2023 will be more than $900 million [1]. More companies are trying to integrate cobot into their production process due to its many advantages. Therefore, these skills are extremely valuable for students to prepare for them in their future career.

With this project, the Engineering Technology department will procure a tool to educate their students on cobot operation and programming. The project will help introduce students to the integration of multiple devices into a single system. One of the main goals of the project is for students and faculty to be able to easily operate the system. In addition, the system will implement Internet of Things (IoT) 4.0, allowing students to better understand IoT applications in a simulated manufacturing cell. For this project, design team purchased a new 6-axis cobot for the ET department. The design team also planned to use the prototype as a demonstration module for high school students, which could help stoke interest from prospective students. Due to the modular design of the work cell, faculty and students will be able to make changes to the work cell to adapt to the change in the automation industry.

In current production operations, there are many menial or repetitive tasks which workers must complete. Over a long period of time by repeating these tasks workers unknowingly cause large amounts of damage to their bodies through repetitive strain injuries. For many of these menial tasks, the work could be automated using cobots. After discussions with a local food company regarding this kind of menial labor, the student design team decided to focus on the automation of such tasks. One of the major menial tasks described by the food company was the issue of palletizing the completed product into shipping boxes. For the industry component of the project, the student design team picked a CNC operation as it is a task where a worker just stands in front of the machine and loads- unloads (machine tending) objects. Despite other ideas that were suggested by faculty advisors, it was decided to pick these two components as they were relevant to ET program needs and doable within the scope of the project.

In the current Engineering Technology curriculum, there is an opportunity to expand knowledge involving the integration of automation into a production system. In order to better prepare students for this subject and the job market, student design team designed and constructed an integrated collaborative robot (Cobot) system which simulates a cobot assisting in CNC manufacturing.

Design Approach

Industry Standards

For this project, student design team researched applicable industry standards which included the ANSI/RIA R15.06-2012 and ISO/TS 15066:2016. These are two important standards that govern industrial robots' design, implementation, and operation. They were developed to ensure the safety of employees who operate or work near robots. Furthermore, they ensure that robots are used effectively and efficiently in industrial settings.

The R15.06-2012 standard is published by the American National Standards Institute and the Robotic Industries Association (ANSI/RIA) [2]. It is a comprehensive guide to the safe design, installation, and operation of industrial robots. The standard covers a wide range of topics, including risk assessment, safety systems, training requirements, and maintenance procedures. The guidelines can be extended to manufacturers, integrators, and users working within the field.

One of the key features of the R15.06-2012 standard is its emphasis on accident prevention. A detailed risk assessment is required before a robot can be installed and operated on. The assessment must identify potential hazards associated with the robot. Which includes the likelihood and severity of potential injuries/property damage. Based on the assessment, appropriate safety measures must be put in place to minimize or eliminate the identified risks.

Moreover, the standard requires the installation of proper safety systems. All robotic fixtures must come equipped with appropriate safety devices such as emergency stop buttons, safety interlocks, and protective barriers. These are in place to actively prevent damage to operators and property during a malfunction.

Importantly, the standard also includes training guidelines for robot operators and maintenance personnel. These guidelines specify the knowledge and skills that users must possess in order to safely and effectively operate/maintain industrial robots. Training topics include robot programming, troubleshooting, and emergency response procedures.

ISO/TS 15066:2016 is another important standard that governs the safety of industrial robots [3]. Published by the International Organization for Standardization (ISO), this standard provides guidelines for the safe use of collaborative robots. These are robots that are designed to work alongside human operators, rather than in separate areas or cages.

One of the key features of 15066:2016 is its guidelines for human-robot interaction. It specifies the maximum force and pressure that a collaborative robot can exert when in contact with a person. Continuing, the minimum distance between the robot and the operator is specified. These guidelines are designed to ensure that collaborative robots are used without putting human operators at risk of injury. Similar to R1115.06-2012, 15066:2016 includes guidelines for risk assessment and safety measures. The standard requires a thorough risk assessment before any collaborative robot is put into operation. Potential risks include operator injuries, active hazards, and property damage. Based on each assessment, appropriate safety measures must be put in place to minimize or eliminate the identified risks.

Lastly, the standard also includes guidelines for safety features and monitoring systems. These safety features may include force & torque sensors, collision detection sensors, and other types of sensors that can respond to unexpected events. Safety monitoring systems may include cameras, laser scanners, or sensors that can monitor both the robot and the surrounding environment.

Competitive co-robot Analysis

The design feasibility for the integrating manufacturing learning module was to integrate cobot arms available on the market regarding the requirement functionalities, performance, features as well as the estimated price for each cobot arm. As a result of our research, the team have identified several potential arms and ensure that the requirement functionalities are met. There are a variety of multi-axis cobot arms available on the market that can feasibly be integrated into the manufacturing process. The first option is the UR3e collaborative cobot by Universal Robots, shown in Figure 1. In terms of project requirements, this robot arm met most of them. It is the world's most flexible, comes with 16 built-in GPIO ports, lightweight table-top cobot to work alongside humans. A few advantages of the UR3e including its speed, interface, and payload. The UR3e weighs 24.3 lbs, has 6 axes, 19.7 in reach, 6.6 lbs payload. Compared to other options, this is a high-quality product with greater precision and accuracy, its small footprint makes it suitable for table-top use and allows it to be used anywhere. The disadvantages of the UR3e are the cost, vision, and additional sensors. The UR3e starts at $24,098, which is a steep offering. Other options are the Kinova Gen3 and the Kinova Gen3 Lite (shown in Figure 1). Kinova launched their first generation of cobot arm in 2009. In 2018 and 2019 they introduced the Gen3 alongside the Gen3 Lite respectively. The advantages of Kinova products are characterized by their light weight, carbon fiber frame, and can be easily lifted with one hand. Compared to other options, they are capable of handling more payload, have a wider reach, and can include an integrated vision system. It is suitable for a lab testing environment. However, the disadvantages of the Kinova robot arm are the lack of a user-friendly interface, the lack of a proper GPIO ports, and pricing. The quoted price for the Gen3 is $29,000, with the Gen3 Lite at $13,500. As a final option there is the uFactory xArm 5 shown in Figure 7. Similar to the other cobot options, the xArm 5 is designed to be used in a factory setting for light weight material handling. It weighs 24.6 lbs, has 5 axes, 27.6 in reach, and 6.6 lbs payload. The advantages of this cobot are it also met most of the requirement features, the cost, as well as the interface. The xArm 5 starts at $5,299. uFactory created xArm studio which allows users to make programming the robot simple. The xArm comes with 32 GPIO built-in ports.

The disadvantages of the xArm 5 include the number of axes, it has the least number of axes compared to other options.

Figure 1: Co-robot market search (From left to right: Kinova Gen3, Kinova Gen3 Lite, uFactory xArm 5Universal Robots UR3e

Work Cell Configuration

The overall layout of the work cell is shown in Figure 2. The main part of the system is the myCobot 320. The cobot is programmed to move the part from station to station to achieve the result. The system flowchart can be found in Figure 6. This explains the general flow of the system and how decisions are made. First, the photoelectric sensor is triggered when a part moves in front of it (Figure 3, a). This trigger causes the conveyor to stop, and the vision camera to take

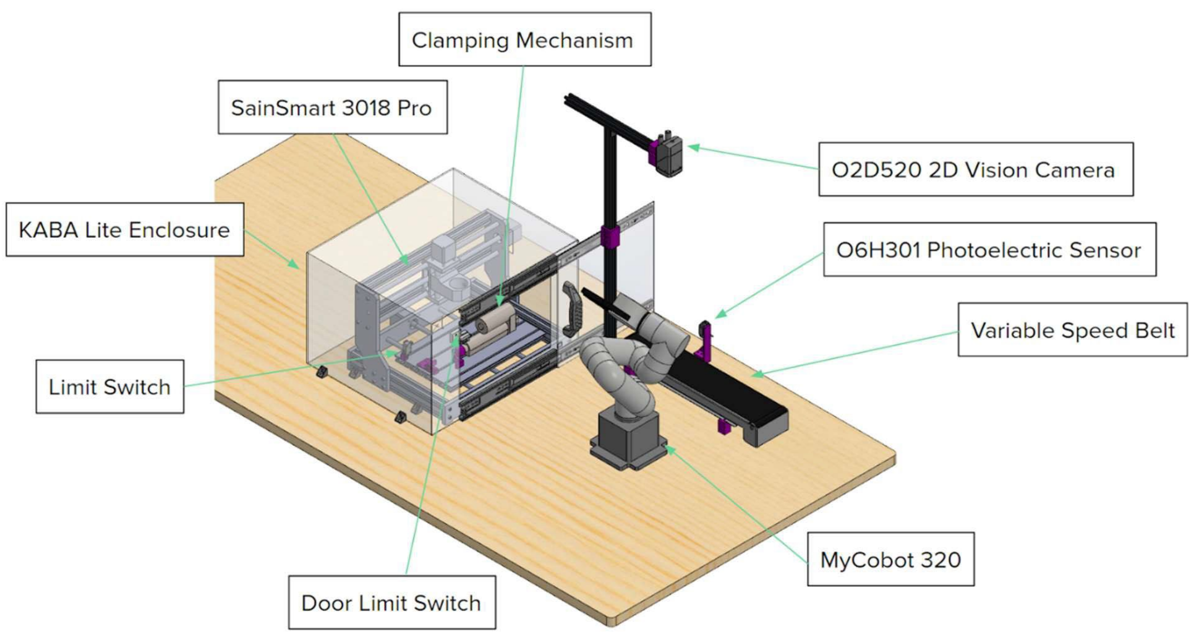
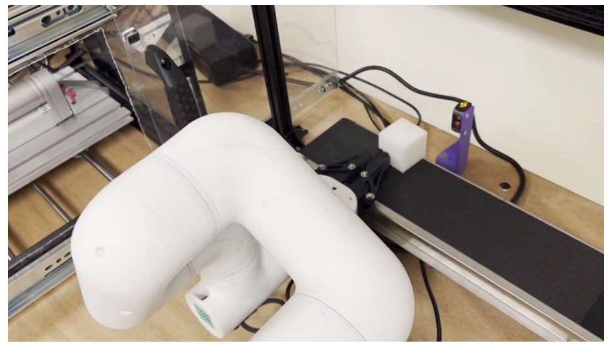


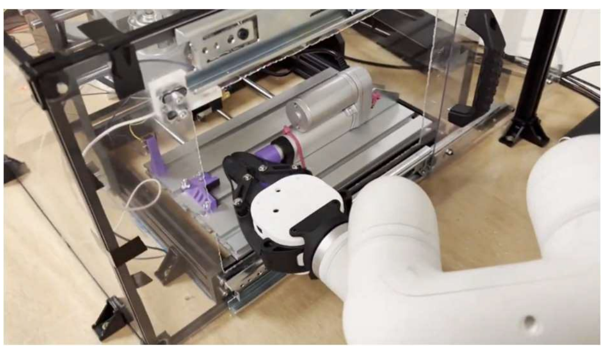
Figure 2: Work Cell Layout

a picture of the part. The camera then sends a signal to the robot, if the part is acceptable, to pick up the part (Figure 3, b). The robot then puts the part in the already open CNC enclosure and the already open clamping mechanism (Figure 3, c). Once the robot closes the sliding door, the clamp tightens, and the CNC program begins (Figure 3, d). When the CNC program is complete, it activates a limit switch to send the signal that the program is done. The robot then opens the door (Figure 3, e), the clamp loosens, the robot picks the part (Figure 3, f) and performs the final step of placing it in the finished part section (Figure 3, g). The robot finally goes back to its home position and the program starts again.

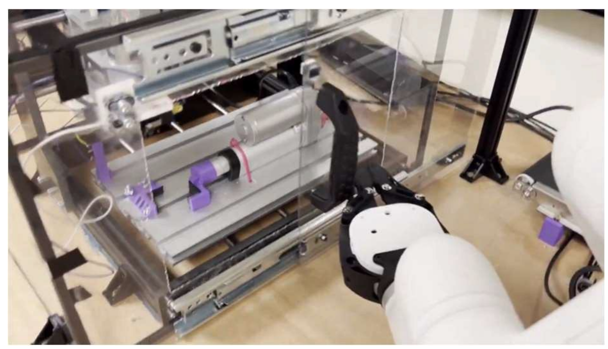
(a)



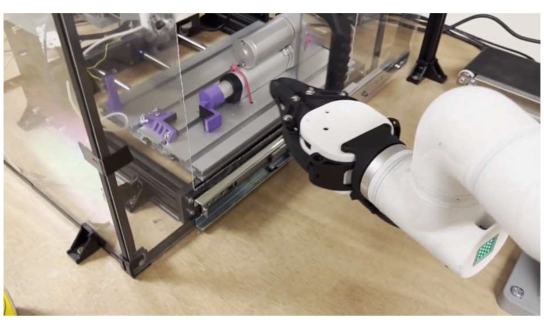
(b)



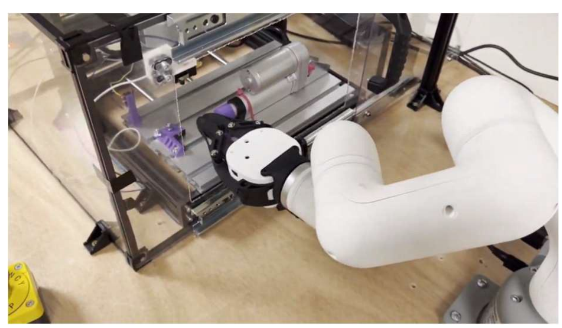
(c)



(d)



(e)



(f)

(g)

Figure 3: Layout Steps

Robot Program and Controller Logic

The main program to control the robot and the overall logic of the system is running on the cobot's onboard Linux computer, the Raspberry Pi 4. The main program also communicates with the Arduino Uno via serial connection. This program is written in Python, and it was created as a state machine (C, Figure 9). This ensured that each step could only be arrived at if the last step was completed, and a signal was received from the Arduino corresponding to that last step (Figure 3).

```
if readCNC() == b'ProgramDone\r\n' and currentStep == 5:
    cobot.send_coords(doorGrabRight,RobotSpeed,0)
    time.sleep(timeDelay)
    cobot.send_coords(doorGrabLeft,RobotSpeed,0)
    time.sleep(timeDelay)
    cobot.send_coords(homeCoords,RobotSpeed,0)
```

Figure 4: Code Snippet from the Main Program

This step in the code is when the CNC sends the signal that it is done with its program. The "and" statement ensures that the current step is correct, set from the completion of the previous step, and that the student design team receive a signal from the Arduino indicating that the CNC program has been completed. This is the framework for which almost all of the steps follow. The only exception is step zero since it is the first step.

To control the robot, the team used the documentation for the "pymycobot" library created by Elephant Robotics [5]. This allowed to move the robot to the predetermined points that were created earlier. These commands also allowed to control the speed of the robot and in what mode it traverses those points. At the end of the program, there is a function to record how many processed parts were created. This can be used for analytics and the potential future implementation of a palletizing operation for the finished products.

For the Arduino program, the code was much simpler compared to the main program (Figure 10).

In the main loop, the Arduino controller is watching the serial port for incoming data. The incoming data are integers sent from the robot program indicating what part of the Arduino system needs to start or stop. If the incoming data matches one of the “if” statements, that action is performed. For example, if the main program sends the Arduino the number 2, the Arduino stops the conveyor. These commands are then followed with a return signal to the main program to show that the operation has stopped. There are also functions to handle the operation of the automatic clamp, the various limit switches, and the signal from the photoelectric sensor.

Robotic Arm and Gripper

One of the critical components for the work cell to function is the myCobot 320 Pi. It is a robotic arm manufactured by Elephant Robotics in China. Based on budgetary limitations, this product was chosen for the work cell. It notably uses 3D printing for the chassis which brings the overall price down. As part of the order package, the robot arrived with an adaptive gripper. Which will be used for translating the Delrin cubes to their required location. The robot is controlled using a Raspberry Pi 4B, and is capable of understanding drag & teach, myBlockly, and Python as programming methods.

The cobot can reach the conveyor belt and CNC workspace and has a work radius of 350 mm. Additionally, it has a working payload of around 1 kg and the manufacturer listed 0.3 mm of repeatability between set coordinates. The adaptive gripper has a clamping force of 1 kg, an actuation range of 0-90 mm, weighs 350 g, and communicates with the arm through a serial interface. The weight of the Delrin cubes and gripper are not concerning, as they each weigh roughly 200 g, which is still within the carrying capacity of the arm.

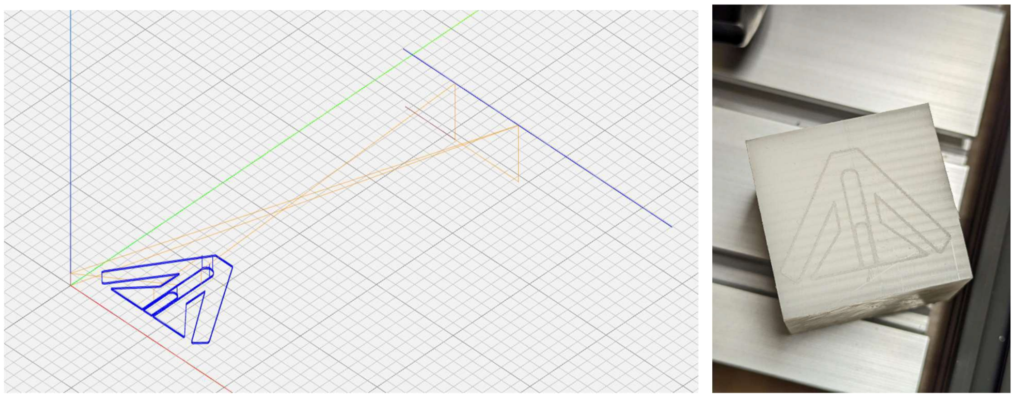
However, there are some notable issues with the cobot itself. The software support out of the box was missing, and online customer support was needed to resolve the issues. Validation testing brought up additional problems with the repeatability of the robot. The team tested this using a ruler and measuring how far off the robot is when translating back and forth through preset coordinates. It was observed a ±1 mm repeatability, with an accuracy of 0-50 mm. Lastly, the gripper was disabled towards the tail end of the prototype assembly. The serial port was giving out constant high signals, and the software on the Raspberry Pi was unable to communicate with the port.

CNC

To simulate the work cell’s manufacturing process, the SainSmart/Genmitsu 3018 Pro CNC router was utilized. It has a small form factor, which helps save space on the table. The spindle motor and carving bits are more than capable of cutting Delrin. The CNC controller can interface on a PC using the open-source, GRBL drivers. Additionally, there is an offline controller provided that reads from SD cards.

The overall frame of the router is comprised of aluminum. V-slot linear rails are used to build the structure and house the 3 stepper motors. For axis movement, ball screw assemblies are used in tandem with stepper motors. Each motor can provide 0.25 Nm of torque. The assembly also provides anti-backlash when running a program. The carving bits provided are v-tipped with a 20-degree angle, and 3.175 mm diameter. The overall frame covers a 400x330x240 mm zone, with the workspace covering 300x180x45 mm. The work area has enough room to house a Delrin cube and linear actuated clamping mechanism.

To program the CNC machine, Easel was used to create a general layout of the logo. Once the layout was made, the specifications were then added to ensure proper operation of the router. The cutting parameters were specified as 12 in/min (304.8 mm/min) feed rate, 5 in/min (127 mm/min) plunge rate, 0.01 in (0.254 mm) depth per pass, 0.0295 in (0.7493 mm) cut depth, and 8000 RPM spindle speed. They were then exported with the G-code from Easel (Figure 11). NC Viewer is then used to verify tool path accuracy (Figure 4). Next, the code was imported into Universal G-code Sender (UGS). This software is compatible with GRBL operated CNC controllers, so it was capable of interfacing with the cnc machine.



(a) (b)

Figure 5: a) NC Viewer Tool Path Simulation b) Finished CNC Cut

The CNC machine ran the code without many problems, and the logo was cut correctly (Figure 5). However, creaking was observed during long movement phases and high vibrations. The parts that connected the steppers to the ball screw assembly had to be tightened as they slipped out during initial tests. Furthermore, the offline controller is not capable of recognizing certain G-code commands, which is why there are extra movements to re-home the spindle after a certain amount of time. It was suspected that the controller would power down the motors during the G4 (pause) command and would signal the program's completion prematurely. This may also be the reason why the controller does not remember the work-home position. There are a lot more observed limitations when working with the offline controller than using a PC interface like UGS.

Vision System

While automation in the workplace has the potential to increase the productivity of a given task, a lower presence of human interaction may lead to otherwise obvious flaws in the process going unchecked. This could range from improper workpiece input to even a damaged component of the work cell still operating. To mitigate such flaws, machine vision has developed methods to act as a quality assurance step to prevent visible irregularities that could cause damage to the overall system (Figure 12, and Figure 13).

The project work cell implements an ifm O2D520 2-dimensional object recognition sensor to evaluate the quality of the workpiece to be transported by the robotic arm and processed by the CNC machine (Figure 14). The purpose is to determine both if the object being transported on the conveyor is the Derlin cube of the correct proportions (1.5in width x 1.5in length) but also that there are no flaws or obstructions that would prevent the part from being considered utilizable. Preventative measures such as quality assurance must be implemented due to the ramifications of an improper part that could damage any of the components being used to process the workpiece, as well as allowing for a rejection that does not hamper productivity by simply allowing the conveyor to continue running until the part is discarded.

The ifm O2D520 can process an inspection image that has been trained with the use of the ifmVisionAssistant software. This software allows the user to determine the functionality of the camera with the use of machine vision trainers, ideal for basic applications such as contour inspection, surface inspection (BLOB function), and I/O interface. The image processing is computed within the camera and can function independently of the ifmVisionAssistant software once the conditions for a passed and failed part have been set.

The first training instruction identifies the area in which the workpiece would appear. The camera itself has been mounted approximately 0.6m above the conveyor belt and focused on the black surface with the white Delrin reference cube on top. Utilizing the 4 infrared LEDs and the contrast editor, the Derlin workpiece can be made into a solid white object behind a solid black background. Allowing the vision processing software to better detect the cube as well as any flaws on its surface. The location of the workpiece was then determined to be right where the belt is expected to pick up the part, so an area was created in the software for the camera to process.

The first training parameter employed was the Contour Function, utilized to define the shape of the cube in reference to all four of its edges and orientation. Because the camera is utilizing a single reference image, the cube selected has been measured to be as close to the specified 1.5in width x 1.5in length as possible. For other acceptable cubes to pass inspection a tolerance of ± 0.1in was taken into consideration, as this deviation still allows for the cube to be properly gripped by the robotic arm and fastened into place by the CNC machine's clamping mechanism. Preventing too much of a grip that would damage the clamping mechanism while also preventing too little of a grip that would dislodge the cube while milling, damaging the CNC. A deviation from the reference image may also indicate that the part being placed onto the belt is not of the shape of a cube at all, as machines in industrial settings would be manufacturing alongside other workpieces that should otherwise be separate from the machine it was intended for, diminishing the risk of mix-ups with this same quality assurance parameter. The orientation of the cube was also given a 15-degree tolerance, meaning the cube can be tilted in either direction by this amount. The gripper itself is large enough to account for the subtle changes in orientation, in which it would self-correct the cube when gripped.

The second training parameter featured is the BLOB Function, which processes the pixels on the face of the object in order to determine if there are any flaws or obstructions on all white faces. Training this feature would require just the face to be completely white when viewed from the camera, as a high contrast allows for the whole cube to be defined into one shape, and anything with a darker coloration would result in the BLOB function detecting it as a flaw. A tolerance was set to accept any piece with 85% of the pixels on the face matching the same contrast as the reference cube, enough to allow for minor blemishes to pass while other larger obstructions would fail if they posed a threat to damaging the CNC.

In order for a part to be considered to pass both the Contour and BLOB parameters must be passed. Upon both parameters being a pass the workpiece the belt is then allowed to stop in the spot the part is to be picked and the initial pick-up program of the cobot has run. Each of these steps is communicated between the camera and the software using the specified I/O wiring configured for the camera (Figure 15).

Wiring the camera and ifm O6H301 photoelectric sensor would require a dedicated power supply to allow operation between 18-30 DCV and 10-30 DCV respectively. A single image is taken of the workpiece once the camera is triggered by the photoelectric sensor when the part comes into range of the sensor. This trigger signal is sent to PIN 2 of the camera once tripped and a photo will then be taken immediately after. The image processing is performed from within the camera and takes approximately 100ms to give a passed or failed result. A passed workpiece gives an output signal through PIN 6, OUT 4 which is wired to the Arduino microcontroller (Figure 16).

Project Challenges

This project faced a variety of difficulties since the beginning of senior design. The myCobot 320 arrived in an inoperable state and many hours were devoted to fixing it. It turned out that the default settings on the Linux operating system caused the UART communication to be off by default. To change this, the memory card of the computer had to be removed, and put into another computer, and that setting was changed in system files.

Once functioning, the myCobot showed issues with accuracy and wobble while moving from point to point, detailed in "Testing and Design Validation" section. This led to recording points many times over to account for the sagging the robot would do on certain operations. The largest issue faced in the project was the gripper malfunctioning. The gripper was tested when the robot first arrived but ended up not working by the final testing phase. This was also troubleshot by the group and the company, but the problem was not resolved in time for the deadline.

The final issues were with the CNC machine's offline controller. The offline controller did not keep the motors powered in-between programs, so the program had to be modified to add extra steps where the machine re-homes itself. Common pause command "G4" caused the program to end instead of pausing the program temporarily. This caused adding extra parts to the program where the spindle moved back and forth to act as the pause. In the end, the design team were able to successfully integrate the whole system, working together.

Design for Environment

This project has been designed with the application of an educational module throughout the decision-making process. Besides the emerging cobot market, cobots are a great tool for education due to their high safety factor and ease of use for programming. The word cobot is a combination of "robot" and "collaborative". Collaborative in this context means that it is safe to be in the same space as operators or students and will not cause injury. Traditional industrial robots need guarding because they cannot detect if they hit something. They also move at very high speeds and weigh a lot. This combination can cause an operator serious injury if there are no safety protocols in place. Cobots are traditionally made of lighter materials, move slower, and have sensors to detect when the robot collides with something. How safe the robot is can also be changed in the settings. The only part of the cell that has guarding is the CNC enclosure. The CNC machine is not collaborative, so it would not be wise to leave it open to wandering fingers and other appendages in an educational laboratory setting.

Approach to Testing and Design Validation

The approach to testing was first accomplished by approving the general concept of the work cell. Once the proper equipment has been purchased, the lead time for robot delivery was spent on the system-level design phase and simulations. Using simulation software like RoboDK; multiple versions of the cell were designed, coded, and simulated.

Most small-scale CNC machines use an Arduino controller, which is easily interchangeable for different purposes. Arduinos allow for a variety of software to be used as well. The open-source software, GRBL, was used as the controller software. The software uses G-Code as its base for sending commands. The controller only needs to process a “Go” signal, mill the placed plastic block, and then send a “Stop” signal. A quick test program was created to test the operation of the mill as it was constructed out of the box. It was important to ensure that the tools can cut through Delrin cubes reliably and that no drifting occurs as the cut cubes are replaced with fresh ones. Additionally, this is a good chance to measure the proper cycle time for the machine, which can be added to the system’s total cycle time.

For the MyCobot 320, RoboDK was utilized for designing various prototypes for where it will be positioned in the cell. The arm’s displacement/movement can be rapidly programmed and simulated through the software. The cobot’s physical movements were then tested to determine the accuracy and repeatability (Figure 17). Upon completion of ten consecutive tests the cobot arm was observed to have a repeatability of ± 1mm, but with an accuracy of between 0-50mm. The cobot also did not have preventions against the backlash that would cause erratic movements before finally reaching each taught location.

With the MyCobot adaptive gripper being used in the work cell operation, a few factors will need to be tested to ensure it works properly. These include the gripping strength, timing of gripping/releasing the Delrin cubes, and stroke measurements. The Delrin cubes need proper support when being transported, as it goes through many changes in direction from the cobot.

With a fast actuator time, the system’s cycle time can be kept efficient. The gripper will also need to fit between the CNC clamping mechanism, which may require testing at various positions to deposit the cube properly before milling can begin. The testing process, however, was postponed due to the malfunctions being detailed previously.

After configuring the O2D520 to the proper working environment of the work cell, the software provided a real-time evaluation of the responsiveness of the image processing. A typical image processing will fall within a 100-150 ms range, ideal for lab conditions when the part is moving in place for the robot to grip the passed part. This delay was also factored into the taught location of the cobot gripper arm.

Devices that will be communicating with a microcontroller will need to understand the signals it receives on demand. These devices are the conveyor belt, photoelectric sensor, and limit switch. The designed code was designed to stop the conveyor belt when the photoelectric sensor detects an object and then start when nothing is detected. Running tests by obstructing the laser of the photoelectric sensor showed instantaneous results.

Overall, upon running the test cube through the work cell in tandem with all components to carve a square through the perimeter of the cube, the system had a cycle time of 314s. This includes a 35s workpiece retrieval from the belt and securing step, a 234s milling process, and a 45s workpiece retrieval from the mill to the finished area and return to home step through. Such a cycle time will vary depending on the milling process, as different paths will yield different results.

Human Factor Consideration

This project is intended to have as much human interaction as possible. With that, It was intend to design a system that considers the operation, interaction, and safety of the working environment. The U.S. Food and Drug Administration (FDA) lists out proper considerations for human interaction with a device[4]. With the considerations in mind, the work cell will have a safe space to work with and reduce most of the human error with operating the cobot. To achieve this, the work cell will be designed with a safe workspace, integrated with a friendly user interface, and instruct users on cobot operations.

To have a safe working environment, the cell will be designed with a safe area for human interaction. Cobots are different from their more industrialized brothers, in that they are designed to be worked with people in close proximity. Their traversal speeds are usually slower, have smoothed edges to avoid cuts, and include sensing technology for contact. This means that whenever the cobot comes into contact with the operator or other people, the robot will stop in its place. The CNC machine will be placed in an enclosure that includes an automatic door. If in an emergency, both the cobot and CNC machine have E-Stop stop switches that are easily accessible.

The cobots are compatible with available software online. The xArm 5 and UR3e come with their own applications for programming. Each comes with programmable General-Purpose Input/Output (GPIO) ports, that allow the robot to integrate with other devices. However, they are also compatible with any PLC if direct connections are not available.

This project aims to teach existing engineering technology students and introduce prospective STEM students. There will be readily available lab manuals and safety statements for the department and students. The instructions are to act as an aid with programming and setting up the cell for demonstration and/or custom workflows. This is to hopefully reduce the human error and accidents caused. This could also increase the effective use of the cell as well and encourage further testing of separate scenarios in a safe environment.

Economic and Project Management Analysis

The total expenditure for the project amounts to $3,701, representing a comprehensive budget that incorporates funds from various sources. To realize This vision, students have secured financial support from multiple entities. Notably, contributions have been received from HAAS, the Department of Education, and ifm, in addition to students own financial commitment. A significant portion of the project's budget is allocated towards the purchase of the robot itself, which accounts for $2,399. This investment is crucial as the robot serves as the central component of the project, enabling students to execute tasks and achieve the objectives efficiently. Additionally, the cost of the gripper, an essential attachment for the robot, is valued at $369. This expenditure is justified by the gripper's crucial role in enabling the robot to manipulate and interact with objects effectively. Moreover, the project requires the use of a CNC machine, which incurs an expenditure of $181. The CNC machine is instrumental in fabricating precise and intricate components, aiding in the assembly and construction of various project elements.

Recognizing the significance of these acquisitions, project advisors took the initiative to procureboth the robot and CNC machines, ensuring that students have the necessary tools and equipment to advance the project effectively. Furthermore, students have been fortunate to receive a generous donation from ifm, which has significantly augmented project's capabilities. This donation includes a 2D vision camera and a sensor, invaluable additions that enhance the robot's perception and data acquisition capabilities. The 2D vision camera enables the robot to process visual information, while the sensor enhances its ability to detect and respond to environmental stimuli. By leveraging the combined resources and funding from HAAS, the Department of Education, ifm Efector company, students have assembled a robust budget that empowers them to pursue the project objectives diligently.

The project process was executed systematically, progressing through various stages over several terms. In the fall term, the initial focus was on brainstorming ideas and selecting a suitable topic for the project. The students dedicated time to researching and gathering information to lay the foundation for the project. As students transitioned into the winter term, their attention shifted towards developing a concrete design concept. During this phase, both 2D and 3D modeling techniques were used to visualize and refine the system. Simultaneously, the required components were assessed and resulted in informed decisions on the parts that needed to be procured for the project. Once the decision-making process was complete, we proceeded to order all the necessary components, ensuring their arrival before the end of the winter term. At the beginning of the spring term, the assembly phase started, bringing together the various components to construct the system. Concurrently, the team focused on programming both the collaborative robot (cobot) and the CNC machine, enabling them to perform the desired tasks accurately and efficiently. However, during the assembly process. By the fifth week of the spring term, the assembly process was successfully completed, allowing to transition to the testing phase. Rigorous testing and fine-tuning were carried out to ensure the functionality and reliability of the system. Finally, in the sixth week of the term, the culmination of the project was reached, successfully achieving the objectives and delivering a complete and operational system.

Societal, Environmental, and Ethical Impact

At the start of the project, the aim was to provide a practical learning environment that simulates an automated process in a collaborative work cell. The intended goal is to provide students with an intuitive look at what sorts of systems would be present in an automated production process, as well as insights on how such systems can be applied for small-scale and artisan production companies that wish to introduce automation into their process. Giving an overview of what this project aims to accomplish, students established a proof of concept in how to procure a product that was not just accessible to Drexel University but to other institutions with interests in practical STEM education.

A main component that was looked at in detail when revising designs was the cobot. While it was expected prior to conceptualizing that a robotic arm would be the bulk of the budget, the priority to obtain a low-cost option was critical. With the cost-effective MyCobot 320, the ability to replicate the building process for the work cell is now more affordable for institutions that do not have access to a research university budget. finalized system aimed to drive the concept of a low-cost work cell but did not take into account the complications a system of such a low procurement cost would entail, such as with lack of backlash prevention and low accuracy.

Conclusions

This senior design project aimed to create a small-scale automation system utilizing a cobot, CNC machine, and conveyor belt controlled through a computer. The system provides hands-on experience as well as bridge the gap between theoretical knowledge and practical application for students in the Engineering Technology program. The compact and portable design allows for demonstrations at fairs and high schools to generate interest in STEM education. The project's concept evolved from a previous proposal for iSwich to palletize ice cream sandwiches, demonstrating the flexibility and adaptability of the design. The project's cost is a key consideration with a decision matrix used to select parts and maintain affordability. Overall, this project has the potential to advance education and innovation in automation and inspire future generations to pursue the STEM field.

References

1. Placek, Martin. “Collaborative Robot Market Size Worldwide 2020-2030.” Statista. Next Move Strategy Consulting, June 24, 2022. https://www.statista.com/statistics/748234/global-market-size-collaborative-robots/. Accessed 31 May 2023.

2. ANSI/RIA R15.06-2012. ANSI Webstore. (2012). Retrieved March 20, 2023, from https://webstore.ansi.org/standards/ria/ansiriar15062012

3. ISO/TS 15066:2016. ISO. (2016, February). Retrieved March 20, 2023, from https://www.iso.org/standard/62996.html

4. Center for Devices and Radiological Health. “Human Factors Considerations.” U.S. Food and Drug Administration. FDA, n.d. https://www.fda.gov/medical-devices/human-factors-and-medical-devices/human-factors-considerations. Accessed 31 May 2023.

5. Robotics, Elephant. “Python API for Mycobot.” GitHub, Elephant Robotics, https://www.github.com/elephantrobotics/pymycobot . Accessed 31 May 2023.

Appendix

A. Decision Matrices & Bill of Materials

		Product Selection (Cobot)							
		A		B		C		D	
		MyCobot 320		Kinova Gen 3 Lite		uFactory xArm 5		UR3e	
		$2,399		$13,500		$5,799		$24,098	
Selection Criteria	Weight	Rating	Weighted Score	Rating	Weighted Score	Rating	Weighted Score	Rating	Weighted Score
Cost	55%	5	2.75	3	1.65	4	2.2	1	0.55
Payload	5%	3	0.15	1	0.05	4	0.2	4	0.2
DoF	5%	5	0.25	5	0.25	3	0.15	5	0.25
Reach	5%	3	0.15	4	0.2	4	0.2	3	0.15
Speed	5%	4	0.2	2	0.1	5	0.25	5	0.25
Interface	10%	2	0.2	1	0.1	4	0.4	5	0.5
Portability	5%	5	0.25	5	0.25	3	0.15	2	0.1
GPIO	5%	3	0.15	1	0.05	5	0.25	5	0.25
	Total Score	4.1		2.65		3.8		2.25	
	Rank	4		3		1		2	
	Continue?	Yes		No		No		No	

Table 1: Robot Decision Matrix

Concept							
		Pneumatic clamping		Hydraulic clamping		Linear Actuator clamping	
Selection criteria	Weight	Rating	Weighted score	Rating	Weighted score	Rating	Weighted Score
Ease of handling	5	3	0.15	3	0.15	3	0.15
Ease of use	15	3	0.45	3	0.45	4	0.6
Cleanliness	25	4	1	3	0.75	5	1.25
Durability	15	3	0.45	3	0.45	3	0.45
Cost	30	3	0.9	3	0.9	2	0.6
Portability	10	2	0.2	2	0.2	5	0.5
	Total score	3.15		2.9		3.55	
	Rank	2		3		1	
	Conitnue	No		No		Yes	

Table 2: Clamping Mechanism Decision Matrix

		Product Selection (CNC)							
		A		B		C		D	
		Vevor 3018		FoxAlien 4040-XE		Benbox 1310		SainSmart 3018 Pro	
		$155		$655		$285		$349	
Selection Criteria	Weight	Rating	Weighted Score	Rating	Weighted Score	Rating	Weighted Score	Rating	Weighted Score
Work Area Size	10%	3	0.3	4	0.4	1	0.1	3	0.3
Control Software	25%	3	0.75	3	0.75	3	0.75	3	0.75
Spindle Power	10%	4	0.4	4	0.4	3	0.3	4	0.4
Cost	20%	4	0.8	1	0.2	3	0.6	3	0.6
Safety	25%	1	0.25	3	0.75	1	0.25	3	0.75
Control interface	10%	3	0.3	2	0.2	1	0.1	3	0.3
	Total Score	2.8		2.7		2.1		3.1	
	Rank	2		3		4		1	
	Continue?	No		No		No		Yes	

Table 3: CNC Decision Matrix

Items	Cost ($)	Qty	Total ($)
MyCobot 320	2399	1	2399
MyCobot Pro Adaptive Gripper	369	1	369
SainSmart CNC 3018 PRO	181	1	181
3018 KABA Acrylic CNC enclosure	70	1	70
12v Power Supply	33	1	33
12v and 24v Relays	25	1	25
Arduino Uno	16	1	16
Stepper motor driver	10	1	10
Motorized Conveyor	106	1	106
DC Motor Drivers (4 Pack)	13	1	13
Limit Switches (10 Pack)	7	1	7
Aluminum Framing	18	1	18
Angle Brackets	23	1	23
2D vision sensor ("ifm" sponsor)	1735	1	0
Photoelectric Sensor ("ifm" sponsor)	149	1	0
PETG Sheet 12" x 12" x 5/64"	6.71	1	6.71
M4 Steel Nuts	3.33	1	3.33
M4 Screws 10 mm long	6.38	1	6.38
Roller Limit Switch	3.95	1	3.95
Unthreaded Pull Handle	10.42	1	10.42
Steel Drawer Slides	24.21	1	24.21
ECO-WORTHY 2 in Stroke Linear Actuator	39.99	1	39.99
		Total	3364.99
		After tax (10%)	3701.49

Table 4: Bill of Material

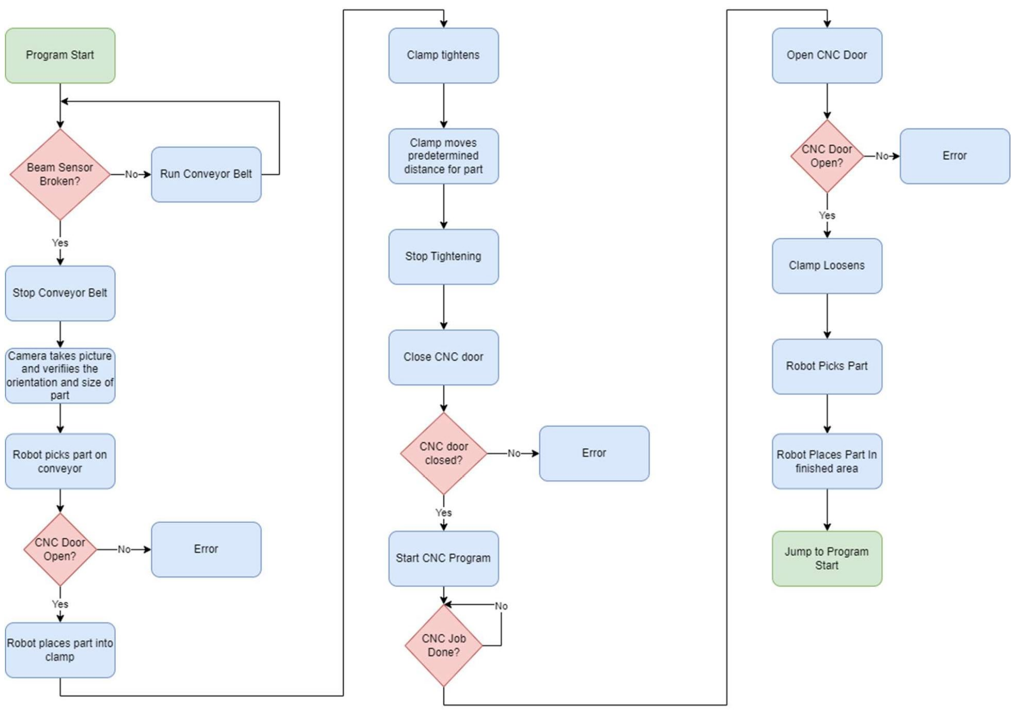


Figure 6: System Flow Chart

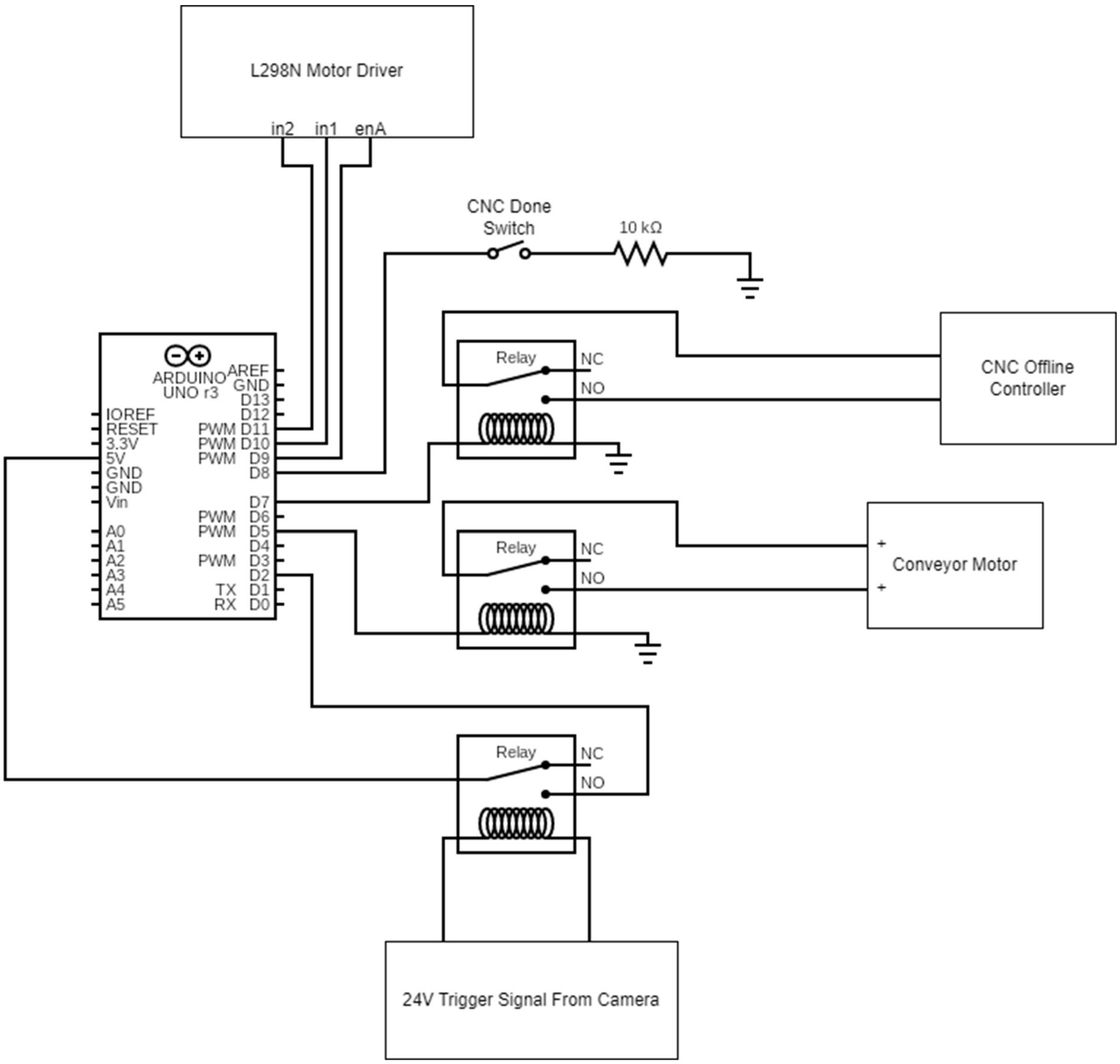


Figure 7: System Wiring Diagram

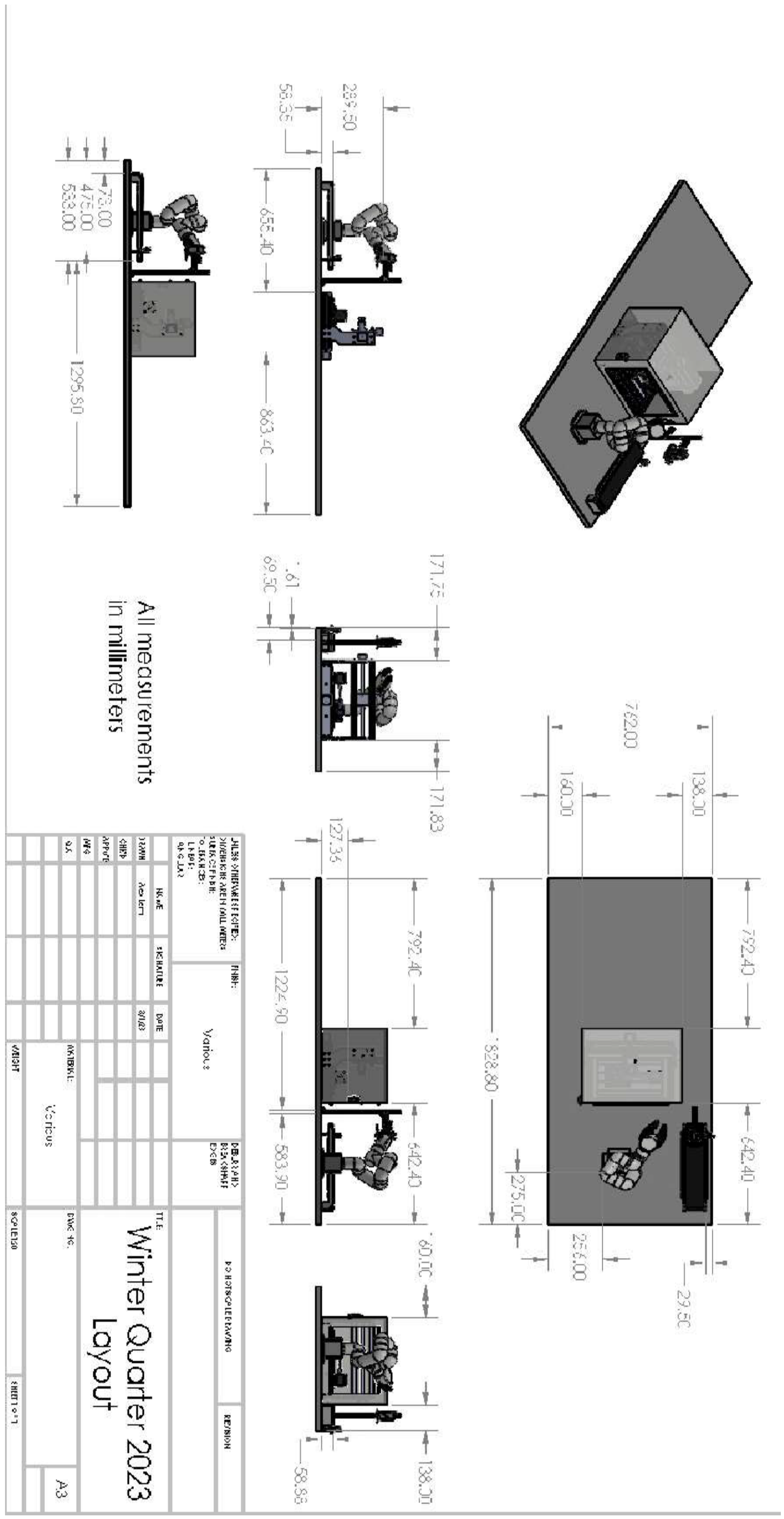


Figure 8: Layout Technical Drawing

```python
import serial
import time
from pymycobot.mycobot import MyCobot
from pymycobot.genre import Coord



#-----------Enter Serial Port Info Here------------
# Initialise serial port for Arduino Uno
SerialPort1 = "/dev/ttyACM0"
SerialBaud1 = 115200
ArdSerial = serial.Serial(port=SerialPort1,baudrate=SerialBaud1,
timeout=0)


cobot = MyCobot('/dev/ttyAMA0')


#important points for robot
homeCoords = [-29.9,-188.4,70.2,88.25,1.59,-90.54]
convApproach = [-127.1,-151.8,67.5,88.28,1.13,-91.93]
convPick = [-149,-161.4,56.4,89.66,-1.32,-91.64]
CNCapproach = [70.5,-239.3,142,95.69,3.13,2.55]
CNCapproach2 = [87.9,-312.3,105.1,88.67,0.65,-4.07]
CNCdrop = [86.8,-308.1,87,91.2,3.17,2.54]
closeDoor1 = [-77.6,-213.9,109.2,91.74,-1.05,-51.26]
closeDoor2 =[145.8,-171.1,109.8,89.87,3.86,-8.08]
closeToOpen = [169,-181.6,238,89.94,-16.17,-8.6]
openDoor1 =[169.4,-189.7,125.1,89.39,-1.95,0.79]
openDoor2 = [-35.3,-210.6,120.7,91.6,0.95,-41.29]
dropApproach = [68.8,-183,77.6,94.02,-2.05,85.3]
drop = [137.6,-184.1,51.6,95.99,-1.28,87.66]




def readArduino():
```

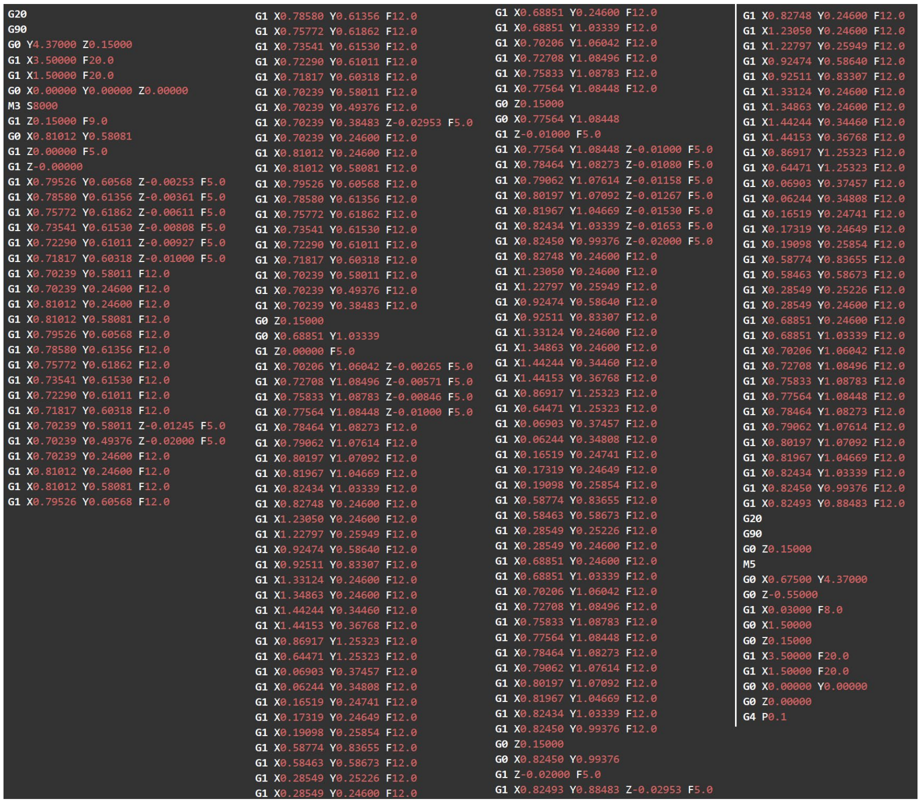
```python
    reply=ArdSerial.readline()
    if reply != b'':
        return reply

currentStep = 0


try:
    go = True
    processedParts = 0
    RobotSpeed = 20
    timeDelay = 2

    while go == True:

        if currentStep == 0:
            cobot.send_coords(homeCoords,RobotSpeed,0)
            time.sleep(2)
            ArdSerial.write(bytes('1\n', 'utf-8'))#conveyor on
            print("conveyor ON")
            currentStep = 1
            print(currentStep)


        if readArduino() == b'BeamBroke\r\n' and currentStep == 1:
            ArdSerial.write(bytes('2\n', 'utf-8')) #conveyor off
            currentStep = 2
            print(currentStep)

        if currentStep == 2:
            cobot.send_coords(convApproach,RobotSpeed,0)
            time.sleep(timeDelay)
            cobot.send_coords(convPick,RobotSpeed,0)
            time.sleep(timeDelay)
            cobot.send_coords(homeCoords,RobotSpeed,0)
            time.sleep(timeDelay)
            cobot.send_coords(CNCapproach,RobotSpeed,0)
            time.sleep(timeDelay)
            cobot.send_coords(CNCapproach2,RobotSpeed,0)
```

Figure 9: Main Program Python Code (Truncated)

```cpp
const int beamSensorPin = 2;
int enA = 9;
int in1 = 10;
int in2 = 11;
int startCNC = 8;
int CNCdone = 7;
int conveyor = 5;


void setup() {
  //set pins as input or outputs. also initializie serial connection
  Serial.begin(115200);
  pinMode(beamSensorPin, INPUT_PULLUP);
  pinMode(enA, OUTPUT);
  pinMode(in1, OUTPUT);
  pinMode(in2, OUTPUT);
  pinMode(startCNC, OUTPUT);
  pinMode(CNCdone, INPUT_PULLUP);
  pinMode(conveyor, OUTPUT);
}

void loop() {

  int x = Serial.readString().toInt(); //read data from serial port

  if (digitalRead(beamSensorPin) == HIGH){
    Serial.println("BeamBroke");
  }


if (digitalRead(CNCdone)==HIGH){ //cnc done signal
  Serial.println("ProgramDone");
  Serial.println("ProgramDone");
  Serial.println("ProgramDone");
  Serial.println("ProgramDone");
  Serial.println("ProgramDone");

 }

//conveyor controls
  if (x == 1){ //conveyor on
    digitalWrite(conveyor, HIGH);
    Serial.println("BeamBroke");
    Serial.println("BeamBroke");
```

```cpp
  }else if (x == 2){ //conveyor off
   digitalWrite(conveyor, LOW);
  }

  if (x == 3){ //3 is the signal for tighten
    tighten();
    Serial.println("TightenDone");
    Serial.println("TightenDone");
    Serial.println("ProgramDone");
    Serial.println("ProgramDone");
    Serial.println("ProgramDone");

  }else if (x == 4){//4 is the signal for loosen
    loosen();
    Serial.println("LooseDone");
    Serial.println("LooseDone");
    Serial.println("LooseDone");
    Serial.println("LooseDone");
  }

 if (x == 5){ //press CNC button to start program
  digitalWrite(startCNC, HIGH);
  delay(300);
  digitalWrite(startCNC,LOW);
  delay(1000);
  digitalWrite(startCNC, HIGH);
  delay(300);
  digitalWrite(startCNC,LOW);
 }


}
void tighten(){
  // For PWM maximum possible values are 0 to 255
  analogWrite(enA, 255);

  // Now change motor directions
  digitalWrite(in1, LOW);
  digitalWrite(in2, HIGH);
  delay(1670);

  digitalWrite(in1, LOW);
  digitalWrite(in2, LOW);
}
```

Figure 10: Arduino Code (C++)



```
G20
G90
G0 Y4.37000 Z0.15000
G1 X3.50000 F20.0
G1 X1.50000 F20.0
G0 X0.00000 Y0.00000 Z0.00000
M3 S8000
G1 Z0.15000 F9.0
G0 X0.81012 Y0.58081
G1 Z0.00000 F5.0
G1 Z-0.00000
G1 X0.79526 Y0.60568 Z-0.00253 F5.0
G1 X0.78580 Y0.61356 Z-0.00361 F5.0
G1 X0.75772 Y0.61862 Z-0.00611 F5.0
G1 X0.73541 Y0.61530 Z-0.00808 F5.0
G1 X0.72290 Y0.61011 Z-0.00927 F5.0
G1 X0.71817 Y0.60318 Z-0.01000 F5.0
G1 X0.70239 Y0.58011 F12.0
G1 X0.70239 Y0.24600 F12.0
G1 X0.81012 Y0.24600 F12.0
G1 X0.81012 Y0.58081 F12.0
G1 X0.79526 Y0.60568 F12.0
G1 X0.78580 Y0.61356 F12.0
G1 X0.75772 Y0.61862 F12.0
G1 X0.73541 Y0.61530 F12.0
G1 X0.72290 Y0.61011 F12.0
G1 X0.71817 Y0.60318 F12.0
G1 X0.70239 Y0.58011 Z-0.01245 F5.0
G1 X0.70239 Y0.49376 Z-0.02000 F5.0
G1 X0.70239 Y0.24600 F12.0
G1 X0.81012 Y0.24600 F12.0
G1 X0.81012 Y0.58081 F12.0
G1 X0.79526 Y0.60568 F12.0
G1 X0.78580 Y0.61356 F12.0
G1 X0.75772 Y0.61862 F12.0
G1 X0.73541 Y0.61530 F12.0
G1 X0.72290 Y0.61011 F12.0
G1 X0.71817 Y0.60318 F12.0
G1 X0.70239 Y0.58011 F12.0
G1 X0.70239 Y0.49376 F12.0
G1 X0.70239 Y0.38483 Z-0.02953 F5.0
G1 X0.70239 Y0.24600 F12.0
G1 X0.81012 Y0.24600 F12.0
G1 X0.81012 Y0.58081 F12.0
G1 X0.79526 Y0.60568 F12.0
G1 X0.78580 Y0.61356 F12.0
G1 X0.75772 Y0.61862 F12.0
G1 X0.73541 Y0.61530 F12.0
G1 X0.72290 Y0.61011 F12.0
G1 X0.71817 Y0.60318 F12.0
G1 X0.70239 Y0.58011 F12.0
G1 X0.70239 Y0.49376 F12.0
G1 X0.70239 Y0.38483 F12.0
G0 Z0.15000
G0 X0.68851 Y1.03339
G1 Z0.00000 F5.0
G1 X0.70206 Y1.06042 Z-0.00265 F5.0
G1 X0.72708 Y1.08496 Z-0.00571 F5.0
G1 X0.75833 Y1.08783 Z-0.00846 F5.0
G1 X0.77564 Y1.08448 Z-0.01000 F5.0
G1 X0.78464 Y1.08273 F12.0
G1 X0.79062 Y1.07614 F12.0
G1 X0.80197 Y1.07092 F12.0
G1 X0.81967 Y1.04669 F12.0
G1 X0.82434 Y1.03339 F12.0
G1 X0.82748 Y0.24600 F12.0
G1 X1.23050 Y0.24600 F12.0
G1 X1.22797 Y0.25949 F12.0
G1 X0.92474 Y0.58640 F12.0
G1 X0.92511 Y0.83307 F12.0
G1 X1.33124 Y0.24600 F12.0
G1 X1.34863 Y0.24600 F12.0
G1 X1.44244 Y0.34460 F12.0
G1 X1.44153 Y0.36768 F12.0
G1 X0.86917 Y1.25323 F12.0
G1 X0.64471 Y1.25323 F12.0
G1 X0.06903 Y0.37457 F12.0
G1 X0.06244 Y0.34808 F12.0
G1 X0.16519 Y0.24741 F12.0
G1 X0.17319 Y0.24649 F12.0
G1 X0.19098 Y0.25854 F12.0
G1 X0.58774 Y0.83655 F12.0
G1 X0.58463 Y0.58673 F12.0
G1 X0.28549 Y0.25226 F12.0
G1 X0.28549 Y0.24600 F12.0
G1 X0.68851 Y0.24600 F12.0
G1 X0.68851 Y1.03339 F12.0
G1 X0.70206 Y1.06042 F12.0
G1 X0.72708 Y1.08496 F12.0
G1 X0.75833 Y1.08783 F12.0
G1 X0.77564 Y1.08448 F12.0
G0 Z0.15000
G0 X0.77564 Y1.08448
G1 Z-0.01000 F5.0
G1 X0.77564 Y1.08448 Z-0.01000 F5.0
G1 X0.78464 Y1.08273 Z-0.01080 F5.0
G1 X0.79062 Y1.07614 Z-0.01158 F5.0
G1 X0.80197 Y1.07092 Z-0.01267 F5.0
G1 X0.81967 Y1.04669 Z-0.01530 F5.0
G1 X0.82434 Y1.03339 Z-0.01653 F5.0
G1 X0.82450 Y0.99376 Z-0.02000 F5.0
G1 X0.82748 Y0.24600 F12.0
G1 X1.23050 Y0.24600 F12.0
G1 X1.22797 Y0.25949 F12.0
G1 X0.92474 Y0.58640 F12.0
G1 X0.92511 Y0.83307 F12.0
G1 X1.33124 Y0.24600 F12.0
G1 X1.34863 Y0.24600 F12.0
G1 X1.44244 Y0.34460 F12.0
G1 X1.44153 Y0.36768 F12.0
G1 X0.86917 Y1.25323 F12.0
G1 X0.64471 Y1.25323 F12.0
G1 X0.06903 Y0.37457 F12.0
G1 X0.06244 Y0.34808 F12.0
G1 X0.16519 Y0.24741 F12.0
G1 X0.17319 Y0.24649 F12.0
G1 X0.19098 Y0.25854 F12.0
G1 X0.58774 Y0.83655 F12.0
G1 X0.58463 Y0.58673 F12.0
G1 X0.28549 Y0.25226 F12.0
G1 X0.28549 Y0.24600 F12.0
G1 X0.68851 Y0.24600 F12.0
G1 X0.68851 Y1.03339 F12.0
G1 X0.70206 Y1.06042 F12.0
G1 X0.72708 Y1.08496 F12.0
G1 X0.75833 Y1.08783 F12.0
G1 X0.77564 Y1.08448 F12.0
G1 X0.78464 Y1.08273 F12.0
G1 X0.79062 Y1.07614 F12.0
G1 X0.80197 Y1.07092 F12.0
G1 X0.81967 Y1.04669 F12.0
G1 X0.82434 Y1.03339 F12.0
G1 X0.82450 Y0.99376 F12.0
G0 Z0.15000
G0 X0.82450 Y0.99376
G1 Z-0.02000 F5.0
G1 X0.82493 Y0.88483 Z-0.02953 F5.0
G1 X0.82748 Y0.24600 F12.0
G1 X1.23050 Y0.24600 F12.0
G1 X1.22797 Y0.25949 F12.0
G1 X0.92474 Y0.58640 F12.0
G1 X0.92511 Y0.83307 F12.0
G1 X1.33124 Y0.24600 F12.0
G1 X1.34863 Y0.24600 F12.0
G1 X1.44244 Y0.34460 F12.0
G1 X1.44153 Y0.36768 F12.0
G1 X0.86917 Y1.25323 F12.0
G1 X0.64471 Y1.25323 F12.0
G1 X0.06903 Y0.37457 F12.0
G1 X0.06244 Y0.34808 F12.0
G1 X0.16519 Y0.24741 F12.0
G1 X0.17319 Y0.24649 F12.0
G1 X0.19098 Y0.25854 F12.0
G1 X0.58774 Y0.83655 F12.0
G1 X0.58463 Y0.58673 F12.0
G1 X0.28549 Y0.25226 F12.0
G1 X0.28549 Y0.24600 F12.0
G1 X0.68851 Y0.24600 F12.0
G1 X0.68851 Y1.03339 F12.0
G1 X0.70206 Y1.06042 F12.0
G1 X0.72708 Y1.08496 F12.0
G1 X0.75833 Y1.08783 F12.0
G1 X0.77564 Y1.08448 F12.0
G1 X0.78464 Y1.08273 F12.0
G1 X0.79062 Y1.07614 F12.0
G1 X0.80197 Y1.07092 F12.0
G1 X0.81967 Y1.04669 F12.0
G1 X0.82434 Y1.03339 F12.0
G1 X0.82450 Y0.99376 F12.0
G1 X0.82493 Y0.88483 F12.0
G20
G90
G0 Z0.15000
M5
G0 X0.67500 Y4.37000
G0 Z-0.55000
G1 X0.03000 F8.0
G0 X1.50000
G0 Z0.15000
G1 X3.50000 F20.0
G1 X1.50000 F20.0
G0 X0.00000 Y0.00000
G0 Z0.00000
G4 P0.1
```

Figure 11: G-Code for 3018 Pro

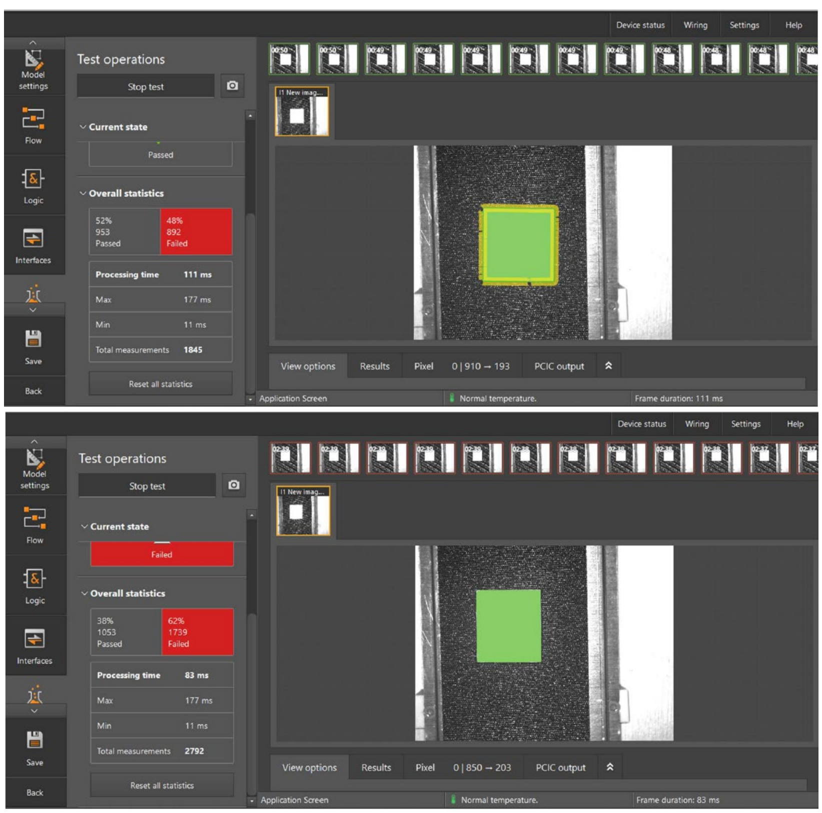
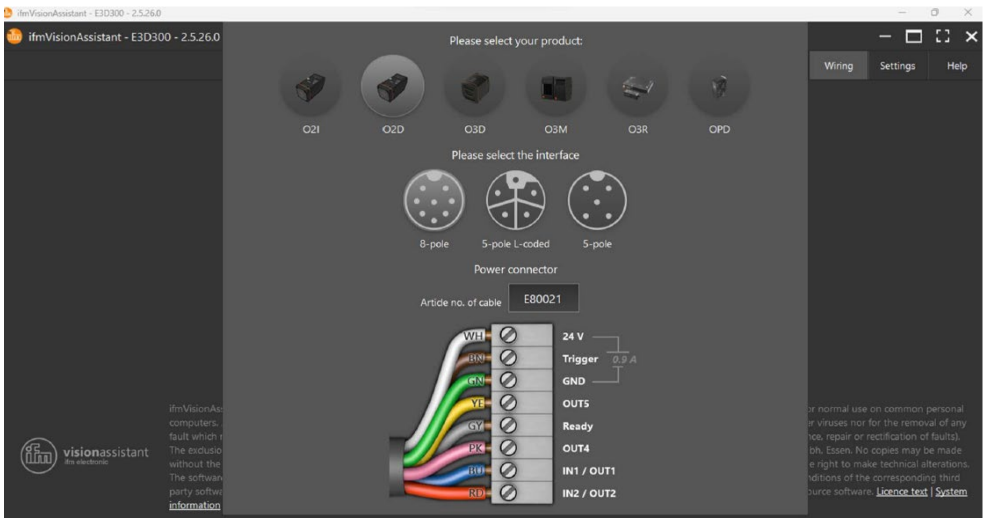
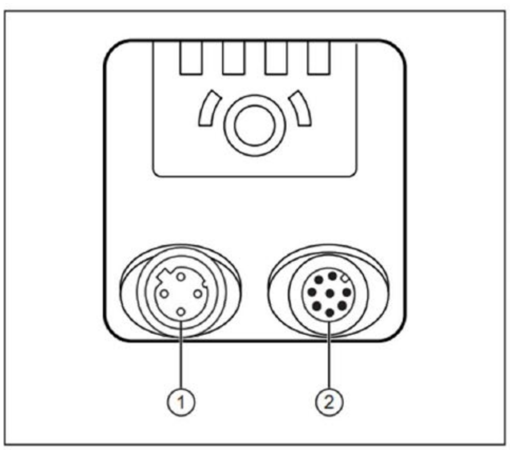


Figure 12-13: ifm Vision Assistant Passed (top) Failed (Bottom) Part on User Interface

Figure 14: ifm O2D520 - Object Recognition Sensor





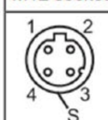
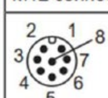
① Ethernet (parameter setting interface, process interface) M12 socket, D-coded, 4 poles	
	1: TD+ 2: RD+ 3: TD- 4: RD- S: screen
② Ethernet/IP (power supply and process outputs) M12 connector, A-coded, 8 poles	
	1: 24 V DC 2: trigger input 3: GND 4: switching output OUT5 5: switching output OUT3 – ready 6: switching output OUT4 7: switching output OUT1 / IN1 8: switching output OUT2 / IN2

Figure 16: ifm O2D520 Pin Diagram

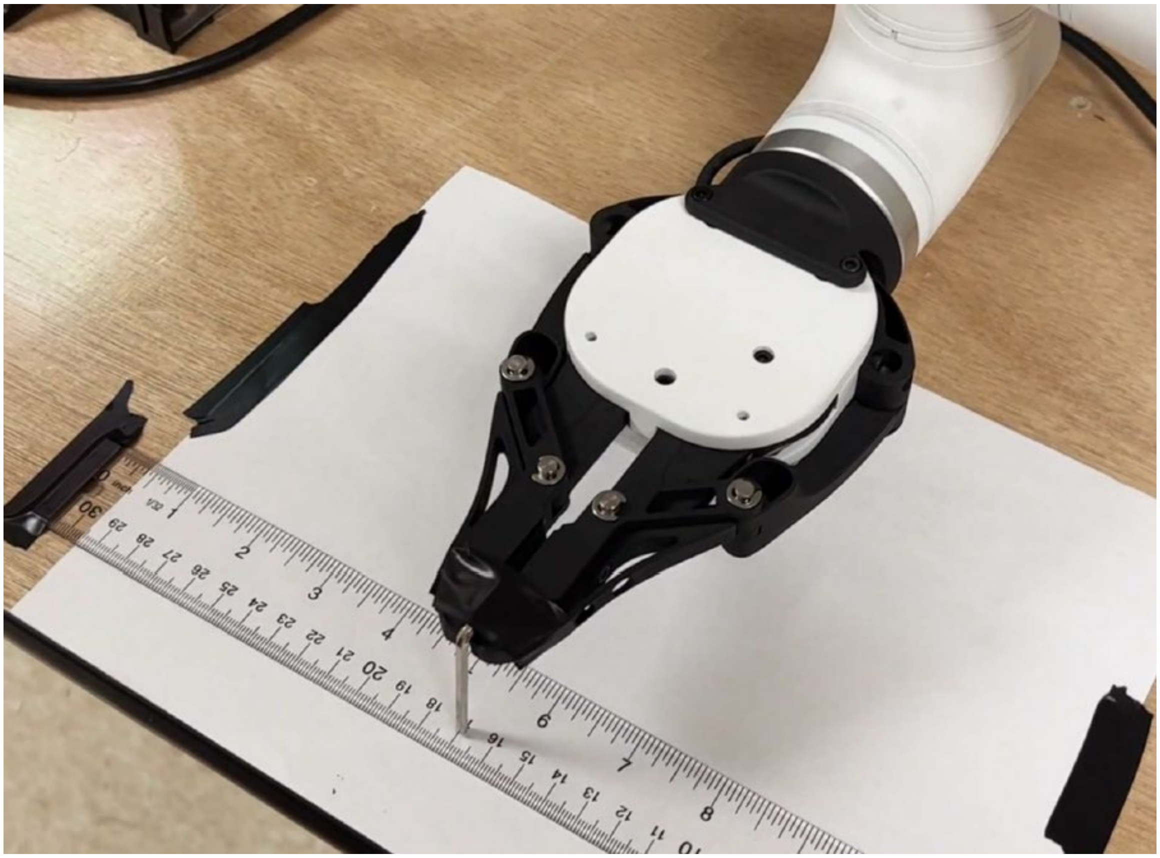


Figure 17: Robot Accuracy and Repeatability Test