Politecnico di Torino

Corso di Laurea in Ingegneria dei Materiali per l'Industria 4.0

A.a. 2024/2025

Sessione di Laurea Luglio 2025

Automated Material Testing

Relatori:
Kai Kallio
Martin Fabian
Matteo Pavese

Candidato:
Lorenzo Bosio

Abstract

In order to exponentially increase the capabilities of the material testing team at Volvo Cars, a deep exploration of automated testing solution was conducted. Firstly, a complete 3-point bending automated cell was developed, able to perform tests according to VDA 238-100 standard and collect results without human assistance. Secondly, possible expansions will be considered, with particular attention on the adaptation necessary to perform tensile tests. Several key challenges are explained, and all the technical solution presented in order to optimize testing efficiency. The knowledge gathered is the starting point for allocating investments into advanced automated testing systems for industrial applications.

Acknowledgments

The initial idea and the completion of the project is thanks to Kai Kallio. He gave extreme support with his vast technical and practical knowledge. Arian Kamal for the immense help when it comes to PLC utilization and software usage. Lucas Scheuer for sharing his knowledge in testing, and being a crucial support in interface with component suppliers. Patrik Träff for the warm welcome and the effort dedicated to making me part of the team. Several companies played a crucial role in providing support and equipment throughout the course of the project. Piab vacuum technology, helped with suction cups and grippers. Robot System product provided the manual tool changer. Zwick Roell and MTS support was extremely reliable. ABB for their knowledgeable and helpful technicians.

Also huge thanks to the professors Martin Fabian and Matteo Pavese, for monitoring the project and helping in maintaining an academic prospective.

Lastly, this would not have been possible without the help and motivation from my family, Alessandro, Rosetta and Francesca, as well as the friends I made along the way Alberto, Giovanni, Leonardo, Renato, Luca, Antoine, Luigi, Matteo, Federico, Francesco all the others.

Contents

List of Figures

List of Tables

Abbreviations and Acronyms

PLC	Programmable Logic Controller
DI	Digital Input
DO	Digital Output
AI	Analog Input
AO	Analog Output
TCP	Tool Center Point
FBD	Function Block Diagram
DIC	Digital Image Correlation
CT Scan	Computed Tomography Scanning

Chapter 1

Introduction

Manually testing the properties of raw materials is a time-consuming and repetitive process. The operator is required to measure each sample one by one, set up the testing equipment properly and evaluate the results. This process is not feasible for baches of hundreds of samples. Manual labour is also affected by human error and data will be subject to slight variations. In such situations the presence of an automatic station makes data collection more precise and in a fraction of the time. The aim of this paper is to guide you through the necessary steps to set up a functioning automatic 3-point bending test cell and discuss the possibilities of automation of other test types, such as tensile.

1.1 Contextualization of the thesis work at Volvo Cars

At the Volvo Cars facility, in Göteborg Sweden, a small material testing team is responsible for handling all the requests from the other departments. Understanding material properties is a crucial step in the research process to develop new material and are required in many simulations for accurate results.

This project is the starting point to understand the capabilities of automated setups, evaluate the challenges along the way and consider whether allocate investments in the future. The vision is to have an advanced laboratory able to test several different sample types and collect data in the most efficient way.

1.2 Aim of the project and evaluation parameters

The project aims at understanding and evaluating the challenges in having an autonomous laboratory. Every step automatized allows the operator to concentrate on higher-value tasks, such as attending a meeting. The steps to be automated are the sample measurement, the testing phase and data collection. Some steps will still require manual labour. These are obtaining samples, labelling them with QR code stickers and analysing the obtained data.

The project is divided in two parts. Firstly, a complete bending test cell is to be developed. 3 point bending was chosen for the easier manoeuvrability of the samples, especially after the testing phase. In doing so the goal is to understand what the main challenges are and the component needed. This system will then be evaluated by obtaining real data and comparing it with results obtained manually.

Secondly, an exploration of the required changes for the tensile test can take place. The goal is to adapt the system to handle both types of test. This part is evaluated by the variety of testing types that can take place.

1.3 Analysis of already existing systems

Materials can be described using a variety of testing procedures, depending on the sort of information needed and the intended use. Destructive testing techniques are commonly used to determine mechanical parameters such as yield strength and Young's modulus [2],[3].

Automated testing becomes particularly useful when dealing with a large number of samples or when great precision and reproducibility are required. Given the specialized nature of such systems, only a limited number are now accessible and thoroughly documented in the literature [4, 5].

Several stations to perform automated material testing have already been developed [6],[7],[8]. This setups thought, don't fully take advantage of the speed capabilities of the cell, designed to be used only for a limited number of samples. This type is more directed towards standardization, rather then throughput and aim at replacing a knowledgeable operator. Larger systems exist for tensile testing [9, 10, 11, 12]. They are custom built for the specific laboratory and it's requirement. Examples range from preserving the fracture surface, to measuring all the dimensions, or even execute multiple tests on the same sample.

Chapter 2

Automated bending system

The following chapter explains the development of the 3 point bending test cell and considerations to be made in order to set up a functioning test cell. The overview ranges from choosing the appropriate components to use, to integrating them to work seamlessly together. Particular attention will be dedicated to ensuring test precision, and validating the accuracy of test results obtained. Finally, potential challenges in system integration, troubleshooting, and optimization of throughput will be addressed, setting the stage for a successful implementation of the bending test cell [13].

2.1 System requirements

The bending test cell had to satisfy the following requirements. Firstly, tracking each sample is needed, avoiding manually written labels and allowing the reconstruction of each sample's history from cutting from the base material to the test data. Preserving the sample after the test is also beneficial, in case the results are outside expectation, visual inspection of the fractured surface can take place. The required precision during execution is moderate. What the robot provides is inside the acceptable limit and can be further increased utilizing all the hardware capabilities. Therefore, the aim will be on productivity. The data obtained from the 3-point bending test is mainly the bending angle. Maximum force applied and estimation of Young's modulus are an added benefit.

2.2 Test setup

The VDA 238-100 standard for three point bending of sheet materials is widely adopted to assess local formability [14],[15]. The standard specifies geometric constraints, including the size and shape of the fixture, as well as the punch dimensions. A diagram illustrating the geometry is provided in Figure 2.1. In addition, the standard outlines test execution parameters to use, such as preload, all the stopping conditions and speed settings to be utilized at each step of the process.

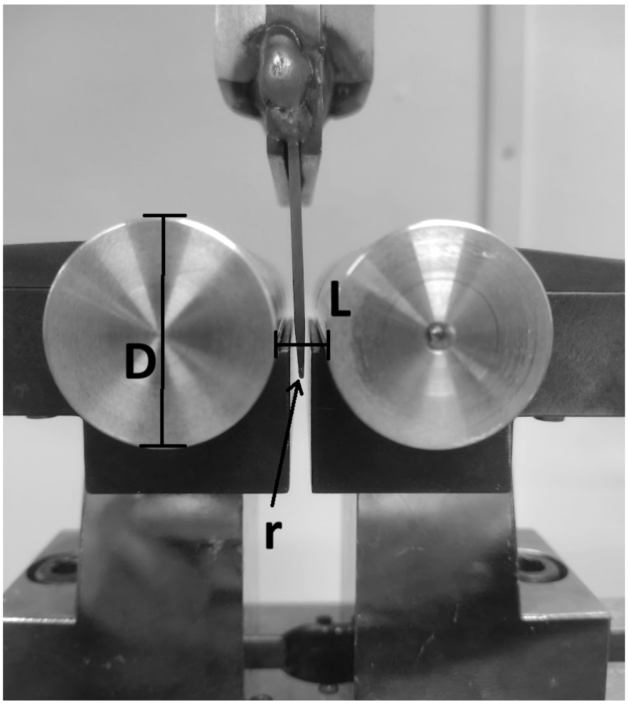

Figure 2.1: Geometry used according to VDA 238-100

The 3 point bending test is performed with 60 mm×60 mm square-shaped samples. Thickness is a crucial parameter in bending and will therefore be measured more accurately for each sample individually during the testing cycle. The distance between the rollers L is a function of the sample thickness t_m. The available fixture used does not allow robotized changes and will therefore have to be fixed in position. Since all samples have a nominal thickness of approximately 3 mm the roller distance is set to 6.09 mm. This decision has also been taken to not introduce variations with the reference data in the results (see Section 2.13). The rest of the parameters are according to the standard and precise values are shown in the Table 2.1 below.

Symbol	Description	Value [mm]
r	Punch radius	0.4 ± 0.02
D	Roller diameter	30 ± 0.01
L	Roller distance	6.09 ± 0.1
b	Sample width	60 ± 2
l	Sample length	60 ± 2
t_m	Sample thickness	Measured for each sample

Table 2.1: Test parameters

2.3 Layout

For organization purposes, a sketch of the layout is necessary. The whole system will have to be placed inside the robot arm reach of 950 mm. The layout, shown in Figure 2.2, includes several key areas, each fulfilling a distinct role within the automated cycle. A pick up station is the location where samples are stored and where the robot grabs one sample after another. The QR code station is where sample recognitions takes place through a QR code sticker. Measurement station is dedicated to evaluating the sample's thickness. At the testing station the sample is positioned and test execution can take place. Finally the disposal station is where the sample is deposited after the testing has occurred.

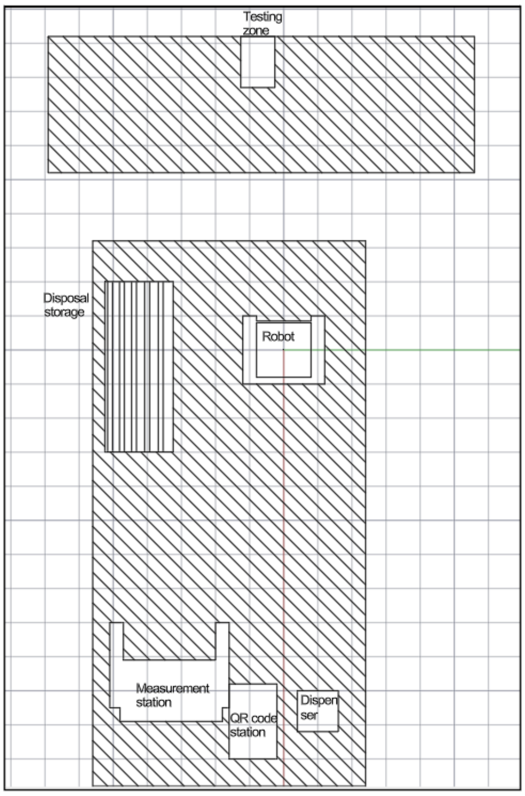

Figure 2.2: Layout plan showing all stations and their relative position.

The layout is in scale, to have a clear idea of the available space for manoeuvrability and the overall appearance of the cell.

2.3.1 Coordinate systems

Several coordinate systems exists and can be either fixed or moving. The main one is the world coordinate. It is fixed and placed in the centre of the robot base. It's the one being used for all object position in the x-y plane represented in Figure 2.2. This way it is easy to access the coordinates of a certain point of station relative to the robot. This same coordinate system is being used for the virtual Robotstudio setup.

The TCP coordinate system is instead a moving system that stays consistent to the tool centre point and is used for tool position when reaching targets. When defining a new tool, a new set of TCP coordinates is created in the specified location. The Work object coordinate system also exists. It is used in case of operations done on and around the work object, in order to stay consistent to it, while ignoring everything else. This has not been necessary to implement.

2.4 Robot setup

The robotic arm plays an important role in moving the sample between the station and ensuring smooth operation can take place. It is managed by the robot controller, which executes the program, sends and receives signals, and deals with errors. At the end of the arm a tool, controlled by the robot, is used as interface with any work-objects, in this case being the test sample itself. This final component of the robot has to be custom made for the application and then defined in the program. It is discussed in Section 2.4.4.

2.4.1 Robot specification

The robot used is an ABB cobot (collaborative robot) CRB 15000 [16]. It has an arm reach of 0.95 m and 6 degrees of freedom. The absolute position accuracy of 0.1 mm ensures the required precision is reached. The maximum payload of 5 kg is not a limiting factor, nor is the maximum speed of 2.2 m/s of the TCP (Tool Center Point) [17]. A visual representation of the robot arm is shown in Figure 2.3.

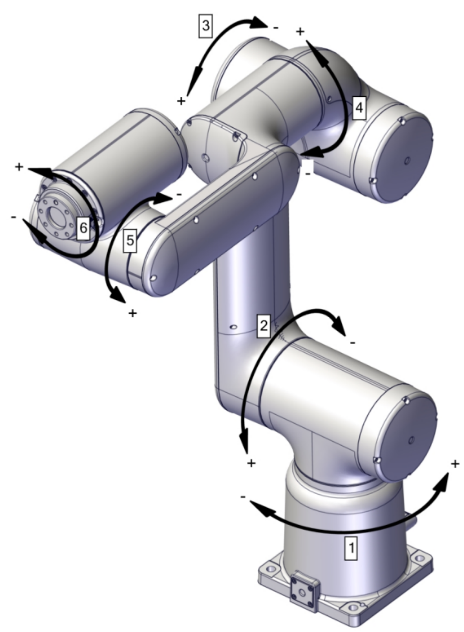

Figure 2.3: Robot schematic [1]

Being a cobot, it has force and torque sensors to stop movement in case of contact with a human or an unexpected object. This feature will also be used for force dependent operations.

The available signals are 16 DO, 16 DI, but no analog input or output is available directly in the robot controller. The robot includes a FlexPendent [18] as interface directly with the robot controller. It is used to visualize uploaded programs, monitor execution and perform jogging operations.

2.4.2 Safety configuration

The robot requires safety configuration in order to operate properly. This settings are implemented in the form of safety zones which enforce either speed or force limitations and need to be tuned correctly in order to avoid encountering any risks.

Beyond zone configuration, comprehensive safety involves performing a risk assessment in compliance with industrial safety standards such as ISO 10218 [19],[20] for industrial robots and ISO/TS 15066 [21],[22],[23] for collaborative applications [24].

2.4.3 Programming

The robot program is written in RAPID [25], a high level language developed by ABB specifically for robots. The compatible computer program is Robotstudio, developed by ABB as well, which allows to freely modify the code, model the station in a 3D environment and simulate the current setup. Points of interest have to be defined at the beginning alongside all tool data and variables [26, 27]. Obtaining robot target can be done by jogging or the lead through functionality of the cobot, as well as defined manually. It's important that all target not only clarify the position in space, but also the orientation of the tool in that position.

Figure 2.4 shows the 3D modelled station inside RobotStudio.

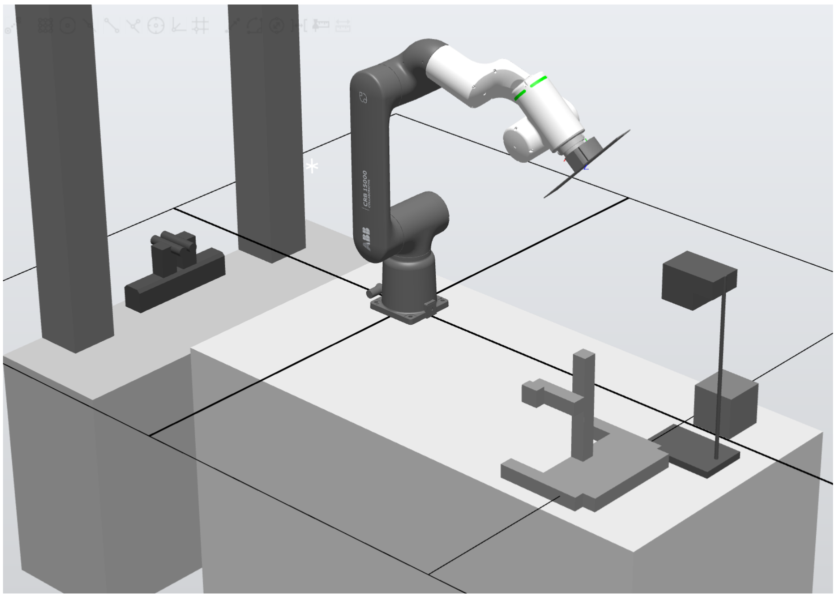

Figure 2.4: Modelled station implemented in RobotStudio.

At the beginning of the programming process, key points of interest, as well as all necessary tool data and variables, must be defined. The robot targets can be specified either through jogging, using the "lead-through" functionality of the robot, or manually. It is crucial that each target not only defines the robot's position in space but also the orientation of the tool at that position.

The program is also responsible for overseeing and altering all signals states in the controller and determines the pace of the whole cell by monitoring the measurement process and test execution.

To facilitate debugging and improve code understanding, the use of functions is essential. By invoking functions only when necessary, it is possible to reuse, adapt, and organize code sections more effectively. The program is executed cyclically an indefinite number of times unless otherwise specified. See Appendix C.3 for the full program.

2.4.4 Tool

The tool, or end effector, is an extremely important component in the robot setup. It is fixed at the end of the robot arm and is the interface between the work object, test samples in this case. It is composed of two components, the tool exchanger and tool head.

The tool exchanger is a small cylindrical component which itself is composed of two parts. The tool attachment is fixed and attaches directly to the end of the robot arm. It receives all electric signals from the controller and air tubes. The tool exchanger instead has the role of passing all signals and air connections to the tool head, while also being detachable. The specific part used was provided by Robot system product, Coboshift tool attachment MTA18 and Coboshift tool changer MTC18, which allows a total of 6 electrical connections alongside 4 air ducts [28]. The tool exchanger used is manual, but automatic solutions exists as well.

The tool head was designed in a 3D cad software to fit onto the tool exchanger with 4 screws. The developed solution is composed of two sides (see Figure 2.5). The first houses two suction cups used to handle the samples before they are tested. The second manages the bent samples after the test has been executed with a vacuum gripper. The suction cups, vacuum gripper, and vacuum controller were all provided by Piab. Modelling the tool head is essential in case of 3D printing requirements. It is beneficial in all other cases to get accurate dimensions, import it in Robotstudio and adapt the safety configuration correctly. Figure 2.5 below shows both the 3D CAD model and the corresponding physical object.

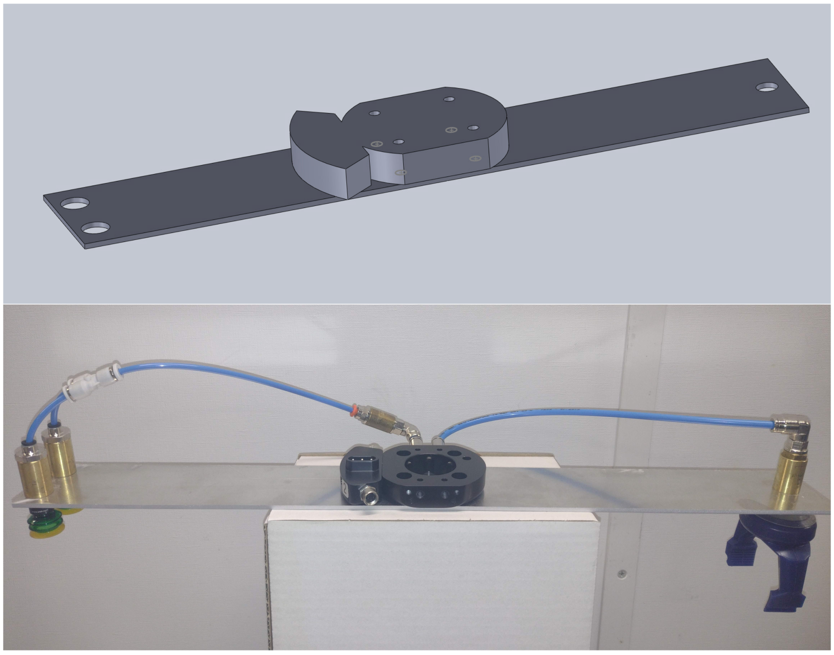

Figure 2.5: Virtual model (above) and physical tool (below).

2.5 Pick up station

For correct usage of the testing cell, the operator is required to cut, label and position the samples to be tested in the storage location before turning the system on. As storage, several solutions can be utilized, but the simplest is the most effective. Samples need to be stacked on top of each other in a specific location recognized by the robot. All successive operation are carried out without further input.

The x and y position of the sample stack needs to be defined, as the robot relies on a searching routine to detect the correct vertical distance. Once the appropriate proximity is reached, the sample is lifted off by the negative pressure in the suction cups, even before coming in full contact, and is securely attached to the tool. Although this approach results in some loss of positional precision, it is acceptable at this stage, as high accuracy is not critical during the initial pickup. However, adjustments must be made to eliminate this uncertainty before placing the sample onto the test fixture, where precise positioning becomes essential for reliable test execution.

In case the sample stack is too high, a multiple stack approach is possible. Using the same principle stack number 2, 3 etc are processed correctly. When starting the program for the first time it's required to input the number of samples in each stack, so that counting the number of cycles executed, it's possible to process every sample.

2.6 QR-code station

The QR code station is composed of a camera and a space for the robot to show the sample to the field of view. The camera is then connected to the computer running a Python script continuously. The purpose of the code is to analyse frame by frame the camera feed, recognize any QR codes present and decode it through the OpenCV library [29]. The QR code contains informations regarding the previous operations each sample has been through. For simplicity purposes it contains the sample number, but can be expanded if required. Each time a new QR code is detected, the code creates a new excel sheet containing introductory column labels and the sample number in the first row. Each cell will then be filled with correct information, already converted into the appropriate measurement unit.

The most critical parameter to adjust, to ensure smooth operation, is the focus point of the camera, as QR-codes not in focus might not be recognised by the camera. Although this is a rare occurrence, placing the camera at 15 to 20 cm from the ground surface can help minimize errors.

2.7 Measurement station

The measurement station, shown in Figure 2.6, is where thickness is evaluated. That is a important parameter in the bending test and needs to be precisely determined. As for the other dimensions, they are not crucial for the bending test, and will not be measured for each sample individually, but an estimation will be given. The measurement process involves calculating the distance difference between when the sample is present and when is not. This way thickness values are extrapolated. To do so, a laser is pointed downwards and a detector is able to evaluate distance based on triangulation angle. The device being used is a CP35MHT80 [30] high performance sensor, capable of 0.05 mm resolution. The output is in the form of a voltage analog signal from 0 to 10 V. Calibration of this device was performed with objects of known thickness, to then construct a calibration line (see Appendix B.1) [31, 32].

This station is positioned at an angle. This is because each sample is dropped and guided by gravity to a specific corner of the station. This way it's possible to eliminate any variation of position in the pick up sequence and increase placement precision when executing the test. Grabbing the sample after this stage is a delicate process as precision is required. For this reason force control was used. This robot functionality allows to preset a path and a force goal in the z direction. When the load is reached the program skips to the following steps. The force limit is set to 5N. This way any imprecision in the lifting sequence is minimized.

The method chosen is not optimal in terms of accuracy and cannot take into account curved samples. Solving this issue involves more advanced equipment to take measurements from both sides [33]. This laser measurement solution was selected for convenience and the possibility to be combined with the alignment.

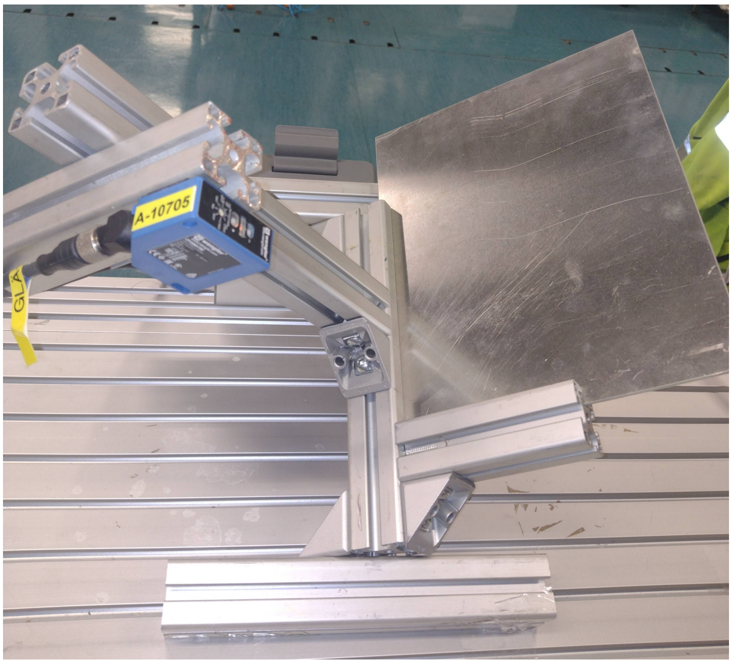

Figure 2.6: Measuring station

2.8 Testing station

The test procedure involves several steps and safety checks in order to proceed. Firstly the sample is placed onto the fixture with high precision. Position confirmation is verified with another laser (similar to the one used during measurement calculation) pointed upwards, ensuring the presence of a test sample. The machine used is the MST Alliance RF/100 [34] (see Appendix A.3) for tension and compression with a Load cell up to 100 kN.

The test procedure starts with a preload phase at lowering the crosshead at 10 mm/min, followed by a main loading phase at 30 mm/min. To accurately determine the bending angle, it is essential to isolate the plastic deformation by removing the influence of elastic recovery. This is achieved by introducing an additional unloading phase following the previously described loading sequence, by moving the crosshead upwards at a rate of 10 mm/min until the applied load approaches 0 N. At this point, the system performs a 5 second hold to allow for stabilization and to record the crosshead displacement, before returning to its initial position. This additional step ensures that the displacement value used for calculating the bending angle reflects only the plastic deformation, excluding any contribution from the elastic one.

Several mathematical models exists to determine the bending angle α_c

in degrees from the crosshead displacement. The standard VDA 238-100 provides the set of equations to use:

$$\alpha_c = 2\left\{\cos^{-1}\left[\frac{W \cdot p - c \cdot (S - c)}{p^2 + (S - c)^2}\right] \cdot \left(\frac{180^\circ}{\pi}\right)\right\}, \tag{2.1}$$

where:

$$W = \sqrt{p^2 + (S - c)^2 - c^2}, \tag{2.2}$$

and:

$$c = \frac{D}{2} + t_m + r. \tag{2.3}$$

Here, S is the crosshead displacement after preload, while the other parameters are described in Table 2.1 (see Appendix C.2, Lines 124-131 for the implementation into the program). Although mathematical models are based on analytical formulas and experimental errors are unavoidable, independent assessments and literature reviews found the angle prediction to be reasonably accurate, typically underestimating by up to 5° [35, 15, 36].

The testing sequence is triggered by a single digital input, while data acquisition is handled through two analog outputs, one for crosshead position and one for applied load. Both outputs are calibrated within the 0–10 V range and are interpreted by the PLC. During the test, the robot tool rotates into position, and once the crosshead returns to its initial state, the vacuum gripper is activated to safely remove the tested sample from the fixture.

2.9 Disposal station

The purpose of the disposal storage, shown in Figure 2.7, is to house all the tested samples in order, while not damaging the surfaces. Following the design philosophy of making things simple, strips of plastic were glued to a base so that samples can be dropped from the top and remain in an upright position. One after the other each sample is placed in a grid separated from the others. Based on the completed cycle number each sample is positioned in the correct location. Completed this step the system resets itself and a new cycle can start.

The number of storage slots in the disposal station is also the limiting factor for the maximum number of samples that can be processed without supervision. Increasing the capacity requires the physical expansion and the corresponding adaptation in the code.

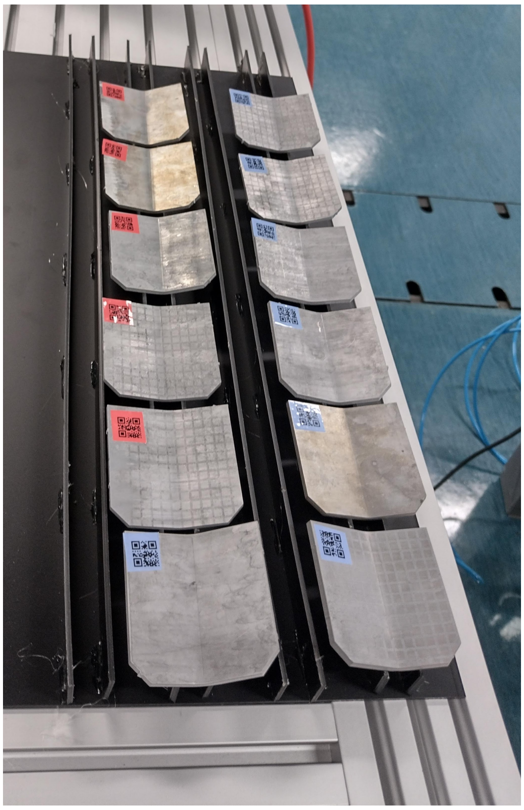

Figure 2.7: Disposal station showing 12 processed samples.

2.10 Signal handling

In order to communicate between the robot controller, the PLC and the other components in the system, several signals were used. Digital inputs and outputs are 24 V switches, either on or off. Analog signals instead can carry any voltage between 0 and 10 V. In the Figure 2.8 below all signals utilized are shown, specifying the entry port when needed. The main components are the robot controller, PLC and PC, exchanging signals with the end effector, MTS testing machine, laser sensors and QR-code camera, while in Table 2.2 the signal use is explained.

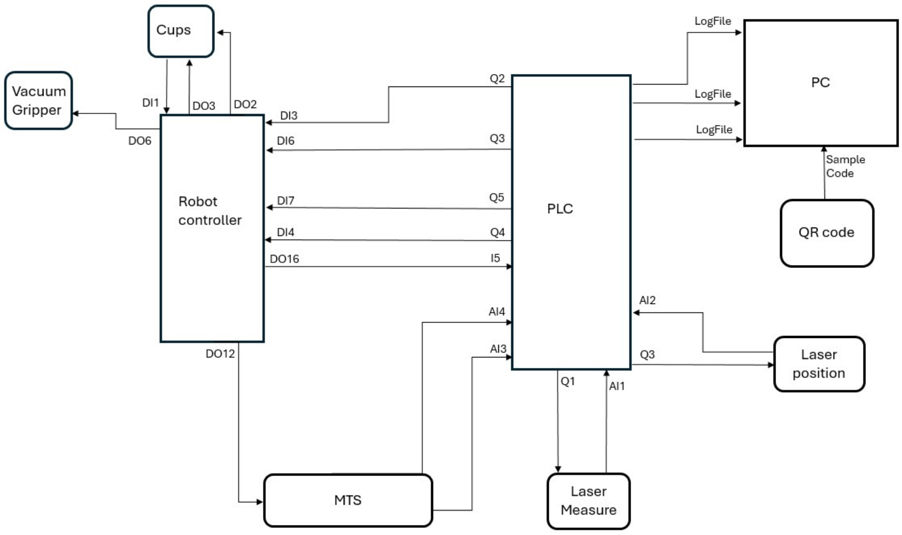

Figure 2.8: Signal map

Name	Type	From	To
Activate vacuum	Digital	Robot controller DO2	Suction cups
Activate blow-off	Digital	Robot controller DO3	Suction cups
Is vacuum on	Digital	Robot controller DI1	Suction cups
Activate gripper	Digital	Robot controller DO6	Vacuum gripper
Measurement is OK	Digital	PLC Q2	Robot controller DI3
Test finished	Digital	PLC Q5	Robot controller DI7
Sample position OK	Digital	PLC Q4	Robot controller DI4
Test type	Digital	Robot controller DO16	PLC I5
Execute test	Digital	Robot controller DO12	MTS machine
Crosshead position	Analog	MTS machine	PLC AI3
Load applied	Analog	MTS machine	PLC AI4
Execute measurement	Digital	PLC Q1	Laser measure
Thickness	Analog	Laser measure	PLC AI1
Do position check	Digital	PLC Q3	Laser position
Position	Analog	Laser position	PLC AI2
Thickness	Ethernet	PLC Ethernet LAN2	PC LogFile
Crosshead position	Ethernet	PLC Ethernet LAN2	PC LogFile
Load applied	Ethernet	PLC Ethernet LAN2	PC LogFile
QR-code data	USB	Camera	PC USB1

Table 2.2: Signal connections

2.10.1 PLC

The requirement of a PLC arises since the robot controller doesn't have any analog signals port. Also real time communication with the computer becomes a possibility, allowing to send data and analyse it with Python. The PLC used was LOGO! 8.3 [37]. The program is written in Function Blocks Diagram (FBD) and then uploaded onto the memory (see Appendix C.1).

The PLC used has some limitations that need to be solved if precise operations are required. Firstly, by default every analog signal is translated into a value from 0 to 1000. This is not enough granularity is some application and is a problem for obtaining precise measurement. Secondly, the communication is only one-way, from PLC to PC by reading all the memory cells every 0.5 seconds. Communication in the other way is not available and a more advanced system is required.

2.11 Data handling

During operation a python script is continuously running in the background (see Appendix C.2). It's role is to select and elaborate data sent from both the PLC and the camera, while also generating the excel sheet with the data gathered. Upon the detection of a new QR code, a new line is created in excel containing the sample code. During operation a CSV file named LogFile is created with a new line every 0.5 seconds containing all the current PLC memory cell values. This includes both digital state (high or low) and analog value. Performing calculation with this values allows to upload the correct reading for thickness, bending angle, max force and slope of the curve in the elastic region.

2.12 Error and safety

In order to ensure that the whole system runs smoothly, not only several safety checks are necessary, but also, in case of encountered errors, the system should reset itself to not stop operation completely.

The most errors encountered were in the robot utilization due to movement or speed restrictions. Safety related errors, for example contact with a human, will stop operation immediately and the system will not restart. Any other mistakes in the robot code are handled by the error handler function (see Appendix C.3, Line 91). This function has the role of trying to resume operation and reset the system for a new cycle to start. The error handler is either called automatically if an error occurred, or manually by raising a custom error when required.

To make sure the sample being processed is suitable, after the thickness measurement the signal DI3 (measurement OK, 2.2) is activated only if the received values is in between predefined limits. This ensures no sample is tested if outside the 1 to 5 mm range.

A position check is executed before performing the test. With the use of a laser the signal DI4 (position is OK, 2.2) is ON only if a sample is present. During the execution of the test limits to the force applied make sure not damage is done and stop operation in case the load limit of 10000 N is reached.

Since the python code is running continuously, errors are displayed on the terminal, but don't stop execution. The most common type of errors are faced during startup of the camera, opening of files and executing calculations.

2.13 Results

The developed cell was evaluated by comparing two datasets. The first one was obtained utilizing the automatic system, while the second one by manually testing samples originating from the same component in cast aluminium. By processing the same material, the objective is to isolate and evaluate only test execution. The system performed as expected and collected thickness measurement, bending angle and peak force for all 12 samples. In Table 2.3 the values of bending angle and peak force are displayed in order for comparison between the two test methodologies.

Method	**Mean Bending Angle [°]**	**Std Dev Bending Angle [°]**	**Mean Max Force [N]**	**Std Dev Max Force [N]**
Automatic	23.3	3.9	5577	686
Manual	24.9	3.7	5491	649

Table 2.3: Comparison of mean and standard deviation

The automatic system being tested showed reasonable accuracy of results, as the variation between the results is minimal. The standard deviation can be interpreted as a measurement of the variance in test execution together with variations due to the material itself [38, 39]. For this reason by testing the same material the goal was to eliminate the material variance and establish the precision in test execution, theoretically better when the automatic setup is used. Although similar, the standard deviation for both bending angle and max force is higher in the autonomous system compared to manual execution. Factors other then repeatability of testing might have played a role. For example, 24 samples were tested manually, compared to only 12 automatically. Also, lower quality of the original material was used in the automatic test. These variables might have driven higher standard deviation, even with similar or better precision during testing.

Cycle time was also calculated while operating the automated bending test cell. Slight variation occurred in the pick up, testing and disposal procedure, but the average cycle time was found to be 176 seconds.

The thickness of the 12 samples automatically processed, was evaluated both manually and by the system itself. Here noticeable discrepancies were observed. The mean difference between the two measurement was 0.19 mm. Considering the nominal thickness of 3 mm a variation of 6% is unacceptable, especially as it plays an important role in interpreting results and calculating flexural strength. This error is entirely caused by the LOGO! PLC converting incoming analog signals in a range of numbers from 0 to 1000. This is not an issue in most application, but in this use case greater granularity is required to enhance precision. The same problem also presents itself when interpreting the other incoming analog signals, such as force and crosshead data. The workaround is to calibrate the output in the machine itself around the interest points, in order to have greater granularity when is needed (see Appendix B.2). This is not a configurable option in the laser.

2.13.1 Imperfections

Aluminium components obtained from casting are subject to several types of internal defects, greatly affecting the mechanical properties. They are grouped into several categories depending on the cause, such as shrinkage, gas-related, filling-related, undesired phases, thermal contraction and metal-die interaction [40]. The main result is the generation of internal porosity, mainly originated by shrinkage behaviour of the material and gas entrapment. They acting as a stress concentrator, lowering the component performance under tension loads.

Detecting these internal defects in cast metal components typically requires expensive equipment and time-consuming procedure. The most common methods are ultrasonic testing [41], X-ray Radiography [42], and Computed Tomography (CT) Scanning [42]. Therefore, identifying a correlation between porosity and another measurable quantity could help determine which samples warrant further analysis.

Dispersed microporosity is known to strongly influence mechanical properties, especially by drastically reducing ductility and strength of the material [43, 44, 45] While the literature is rich for tensile test, little is known about the effect of porosity on bending test results.

In this study, the only data obtained from the automated system were bending test results, including bending angle and maximum force. To investigate potential correlations with porosity, the 12 automatically processed samples were CT scanned and are shown in Figure 2.9. The results plotted in Figure 2.10, were analysed to assess any relationship between porosity and the mechanical test data.

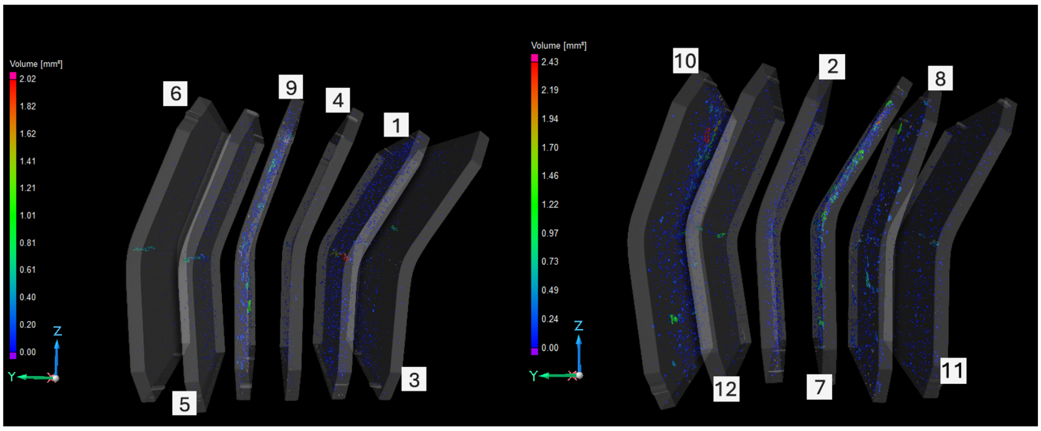

Figure 2.9: CT scan of the samples from 1 to 12.

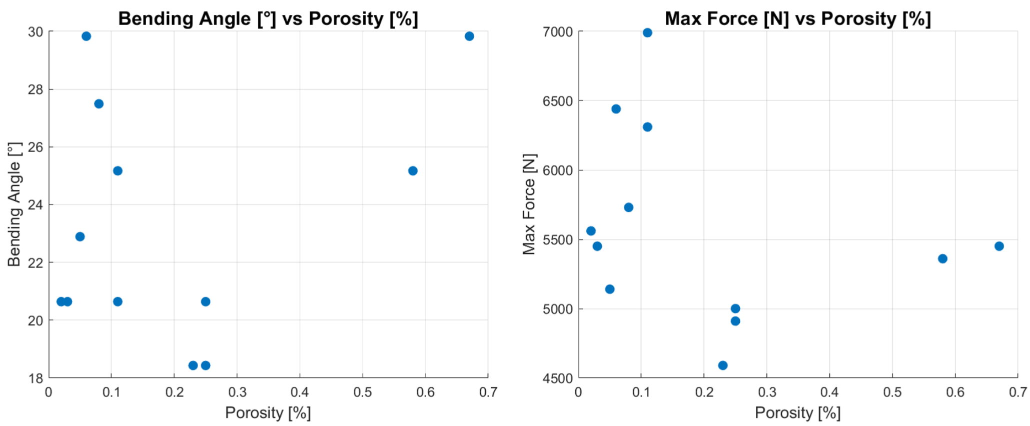

Figure 2.10: Data for bending angle (left) and max force (right).

The correlation coefficient (r value) for bending angle and porosity is 0.3, while for max force and porosity is -0.29. This indicates a vague trend for both cases, but not strong enough to make any reliable predictions. In order to determine any relationship, further evaluation is required with more data points [46, 47, 48].

Chapter 3

Automated tensile system

The system can be expanded in several different ways. One option is to adapt the system to process a wider variety of sample shapes and allow application in a more diverse use cases. Another option is to extend the number of steps performed by the robot while reducing the ones executed by the operator, for example automatic sample labelling. The third option is to expand the capabilities and perform different test types by modifying the system to accommodate different procedures. In any case, the current system should not be compromised and possibly be used at any moment.

Even if the main focus of the project was dedicated to the development of the automated three point bending test cell, other test types were also examined for automation. Since the tensile test provides some of the most useful properties, such as Young's modulus and yield strength [49], it has the highest priority to be automated as well.

Due to the limited time-frame of the project, this second section is less developed compared to the bending test cell. Nevertheless, the progress achieved and the insights gained are presented in the following discussion.

3.1 Requirements

Several key requirement from the bending test are still valid for the tensile test, for example, preserving samples to inspect the fractured surface and tracking with QR-code is not to be neglected. Differences arise when considering that the strain field during test execution is monitored by painting dots on the surface and interpreting data with a DIC software. For accurate results, the surface must exhibit suitable properties such as elasticity and adhesion [50]. For this reason, the surface is extremely delicate and needs to be handled accordingly [51, 52].

3.2 Test Setup

The machine is composed of two grippers, able to regulate the pressure at which to close when the specimen is inserted and the camera for strain field measurement with DIC. There is a specific sequence to follow to ensure proper test results.

First, the sample is inserted into the lower gripper and secured in place. Secondly the testing procedure is started and an approach path is responsible of lowering the upper gripper in place. Thirdly, while the upper gripper is being closed, the crosshead position has to be adjusted to not induce any compression stresses prior to test execution. Once all the conditions are met, it is possible to drive the machine up to the breaking point.

To finish the procedure, the crosshead arrives to a preset position for correct handling with the robot.

3.3 Robot Setup

The robot used is the same as in the previous section 2.4.1. Being mounted on a movable platform, it can be transported in different locations for different test types. This way the bending test cell remains available to use by moving the platform back in place. To execute different test the whole robot code had to change. Firstly, when starting the system for the first time, the user is required to choose between the different test types available (currently bending and tensile). Following this decision the correct section of the code is selected.

The preferred test type is then communicated to the Python script via the PLC, making sure that the appropriate part of the code is executed.

3.3.1 Tool head

Accommodating the new sample shape (see Appendix A.2) requires an appropriate end effector. This component was designed to grab tensile test samples without damaging the painted surface. The padded jaws from Piab (specifically the GRZ 10-10 NPC-P [53]) was mounted at the tip of the end effector, and a distance laser sensor was integrated to monitor the separation between the gripper jaws in real time. The final version of the end effector used in this setup is shown in Figure 3.1.

Figure 3.1: Tool used to handle tensile test samples.

3.4 Robot-Machine communication

The robot has to be able to control the machine grippers during the opening and closing sequence, and also start the test execution at the appropriate moment. This can be controlled by turning on the correct digital signal for a specific amount of time. In my case 3 seconds is enough to securely grip samples. The I/O interface is used to communicate with the machine to start and and then monitor test execution.

3.5 Disposal

The final step of the cycle is the removal of both sides of the broken sample from the machine grippers. This step can prove to be particularly challenging, for the unpredictability of the breaking point along the sample length. To solve this problem, the distance laser sensor is utilized. The robot moves in a trajectory towards the sample lower part until the correct distance is achieved. After the robot secures the part, the lower machine gripper can be opened and the first part disposed in a predefined location. The same procedure can be adapted for the upper section.

Chapter 4

Discussion

The use of a cobot in mechanical testing proved to be valuable, by increasing consistency and reliability of test execution. Although promising, a long development phase is still required to achieve the final goal of a automatized laboratory. The project proved the possibility and potential of robotized testing, as well as highlighting some of the challenges. A separate discussion is provided for the bending and tensile test setups.

4.1 Bending test system

The implementation of a robotic station for automatic 3-point bending was successful and can now be used in every day tasks. The real advantage is in terms of time saved and throughput. The operator spends between 10 to 15 minutes on average to measure a sample, set up the machine configuration, execute the test and finally gather the results. The automatic setup can perform the same tasks in under 3 minutes, while also freeing up time of the operator and be able to work 24-7.

The robot movement are smooth and few to no errors are encountered during operation. Using functionalities such as force control or external sensors increases reliability, allowing the robot to adapt to from case to case problems and dealing with them effectively. The problem relies in the interpretation of the PLC and the subsequent loss of resolution. The use of a PLC is fundamental to interpret analog signals, establish communication and exchange data with the PC. It's an essential component in any automation system for the versatility of programming and possibilities it offers. In combination with python, organization of data is simple and reliable.

4.1.1 Improvements

Although the cell is functional and operating, resources and time would allow more precision, faster execution and more processing capabilities. Possible improvements in order of importance are discussed here.

The LOGO PLC used is a limiting factor when it comes to measurement precision and number of signals available. Another downside of the LOGO PLC, is the unavailability of two-way communication with the computer. By using a more advanced model such as the S7-300, more functionalities are available, and signals could be manipulated through the Python code directly. The quality of the machine itself is also a factor in determining quality of results. For this reason, modern equipment is required to acquire accurate data. To cut down on cycle time, a possibility is to perform operations with the robot when it is idle. During testing, a new sample can be processed by overlapping two cycles. Although an easy implementation, it could face issues during data logging and the current methodology is not suitable. Occasionally, three-point bending tests must be conducted on small or irregularly shaped components with careful consideration. Currently is not advised to perform test on non-standard samples, as unexpected behaviour can occur. Enhancing the system's adaptability to accommodate a wider variety of sample geometries would significantly broaden its range of potential applications. Adding functionalities is an ongoing process. Introducing diverse test types to be executed in line would allow to gather more data on the single specimen, such as hardness evaluation. Moreover, incorporating automated sample labelling could help reduce manual tasks and improve overall workflow efficiency.

4.2 Tensile test system

The robotic setup was moved near the tensile testing machine and adapted to perform both tensile and bending test. Equipped with a new end effector, the system can execute precise movements by continuously measuring the distance between the gripper and any surrounding surfaces in real time. The testing cycle includes a loading phase into the machine's grippers, initiating and executing the test procedure by communicating with the machine, and removing the fractured parts after test completion.

4.2.1 Improvements

Even if the basic testing cycle is functional, the system is still requires development and needs to be refined before being considered operational.

A proper storage and pick-up station must be implemented to enable continuous operation. This station could also incorporate a dedicated mechanism for spraying the dotted pattern to the sample and ensure correct performance of the surface. Additionally, a measurement station should be integrated to capture not only thickness but also the other sample dimensions.

Another critical aspect is data logging. A lot of data is generated during each test in the form of pictures, and managing this information effectively is still a challenge, particularly in automating the calibration of the DIC camera for each new test. Also, no safety features have been implemented yet, meaning any error would halt the system immediately. For this reason, constant monitoring is advised. Even thought the basic operating cycle is functional, this is an ongoing project that still needs refining to be used.

Chapter 5

Conclusion

A fully functional 3-point bending test cell was developed and the ground work for the adaptation to tensile testing laid out. Given the repeatability of the test execution and the precision constantly required, the use of a robotic system is justified. Several challenges still needs to be addressed, but vision of an automatic laboratory is now in the realm of possibility. The hope is that a team of expert, backed up by investments in the field, can finish the work and exponentially increase efficiency.

Bibliography

[1] ABB. Product specification: Crb15000. Technical Report 3HAC077390-001, ABB, February 2025.

[2] Howard Kuhn and Dana Medlin. *Mechanical testing and evaluation.* ASM International, January 2000.

[3] Kyriakos Komvopoulos. *Mechanical testing of engineering materials.* Cognella Academic Publishing, January 2017.

[4] ML Renauld, JA Scott, LH Favrow, MA McGaw, MD Marotta, and DM Nissley. An automated facility for advanced testing of materials. *APPLICATIONS OF AUTOMATION TECHNOLOGY IN FATIGUE AND FRACTURE TESTING AND ANALYSIS, VOL 4*, 1411:16–26, 2002.

[5] ZwickRoell GmbH Co. KG. Robotic Materials Testing: Optimum quality assurance at reduced cost. *ZwickRoell GmbH Co. KG*, June 2010.

[6] W. Lim and H.-K Kim. Design and development of a miniaturised tensile testing machine. *Global Journal of Engineering Education*, 15:48–53, January 2013.

[7] Miguel Hernández-Del-Valle, Christina Schenk, Lucía Echevarría-Pastrana, Burcu Ozdemir, Enrique Dios-Lázaro, Jorge Ilarraza-Zuazo, De-Yi Wang, and Maciej Haranczyk. Robotically automated 3D printing and testing of thermoplastic material specimens. *Digital Discovery*, 2(6):1969–1979, January 2023.

[8] Julia Głowacka, Łukasz Derpeński, Miłosz Frydrych, Bogna Sztorch, Błażej Bartoszewicz, and Robert E. Przekop. Robotization of Three-Point bending mechanical tests using PLA/TPU blends as an example in the 0–100 *Materials*, 16(21):6927, October 2023.

[9] Instron. At6 6-axis robotic automated testing system. https://www.instron.com/en/products/testing-systems/automated-testing-systems/at6-6-axis-robotic-automated-testing-system, n.d.

[10] ZwickRoell. robotest robotic testing system. https://www.zwickroell.com/products/automated-testing-systems/robotest-r-robotic-testing-system/, n.d.

[11] ZwickRoell GmbH Co. KG. Robotic materials testing: Optimum quality assurance at reduced cost. https://www.azom.com/article.aspx?ArticleID=5283, November 2022.

[12] Zwick GmbH & Co KG. Robotic testing of composites. https://www.ien.eu/article/robotic-testing-of-composites/, December 2009.

[13] Cezary Zieliński. *General Robotic System Software Design Methodology.* January 2019.

[14] Ralf Neuhaus and Michael Borsutzki. Plättchen-Biegeversuch nach VDA 238-100. *Materials Testing*, 55(9):654–659, October 2013.

[15] K. Cheong, K. Omer, C. Butcher, R. George, and J. Dykeman. Evaluation of the vda 238-100 tight radius bending test using digital image correlation strain measurement. *Journal of Physics: Conference Series*, 896(1):012075, 2017.

[16] ABB. Product manual – crb15000 (3hac077389-001). https://search.abb.com/library/Download.aspx?DocumentID=3HAC077389-001, 2021.

[17] ABB robotics. Gofa™ CRB 15000 tecnical data. 2023.

[18] ABB. Flexpendant product manual – 3hac093167-001 (rev. a). https://library.abb.com/d/3HAC093167-001, 2024.

[19] Weidong Li, Yudie Hu, Yong Zhou, and D. Pham. Safe human–robot collaboration for industrial settings: a survey. *Journal of Intelligent Manufacturing*, 35:2235–2261, June 2023.

[20] International Organization for Standardization (ISO). ISO 10218-1:2025 – Robotics— Safety requirements— Part1: Industrial robots. https://www.iso.org/standard/73933.html, 2025.

[21] International Organization for Standardization (ISO). ISO/TS 15066:2016 – Robots and robotic devices— Collaborative robots. https://www.iso.org/standard/62996.html, 2016.

[22] Peter Chemweno, Liliane Pintelon, and Wilm Decre. Orienting safety assurance with outcomes of hazard analysis and risk assessment: A review of the iso 15066 standard for collaborative robot systems. *Safety Science*, 129:104832, 2020.

[23] Robin Jeanne Kirschner, Nico Mansfeld, Saeed Abdolshah, and Sami Haddadin. Iso/ts 15066: How different interpretations affect risk assessment. https://arxiv.org/abs/2203.02706, 2022.

[24] Seungbin Moon and Gurvinder Virk. Survey on iso standards for industrial and service robots. *ICCAS-SICE*, pages 1878 – 1881, September 2009.

[25] ABB. Technical reference manual – rapid overview (3hac050947-001, rev.u). https://library.abb.com/d/3HAC050947-001, 2025.

[26] David Marshall. Abb's robotstudio software enables cost savings in robot-programming. *Industrial Robot*, 29(6), 2002.

[27] C. Connolly. Technology and applications of abb robotstudio. *Industrial Robot*, 36(6):540–545, 2009.

[28] Robot System Products AB. Technical description – tool changer atc18/mtc18 (m0121-1). https://downloads.robotsystemproducts.com/manuals/M01211_Technical_Description_Tool_changer_ATC18_MTC18.pdf, 2025.

[29] V. Mathumitha, Krishna Kumar, M. Ganesh Reddy, K. L. Sekhar, and Kuruba Pramod. Enhanced detection and decoding of qr code and barcode using machine learning. *International Journal of Research in Engineering, Science and Management*, 7(4):22–27, April 2024.

[30] wenglor sensoric GmbH. Cp35mht80 laser distance sensor (triangulation). https://www.wenglor.com/en/Sensors/Photoelectronic-Sensors/Laser-Sensors-for-distance-measurement/Laser-Distance-Sensors-Triangulation/Laser-Distance-Sensor-Triangulation/p/CP35MHT80, 2025.

[31] Jorge Santolaria, David Guillomía, Carlos Cajal, José A. Albajez, and Juan J. Aguilar. Modelling and calibration technique of laser triangulation sensors for integration in robot arms and articulated arm coordinate measuring machines. *Sensors*, September(9):7374–7396, 2009.

[32] Gerardo Antonio Idrobo-Pizo, José Maurício S. T. Motta, and Renato Coral Sampaio. A calibration method for a laser triangulation scanner mounted on a robot arm for surface mapping. *Sensors*, 19(8), 2019.

[33] AZoSensors. Thickness measurement with integrated width determination. `https://www.azosensors.com/article.aspx?ArticleID=843`, July 2017.

[34] MTS Systems Corporation. *Insight Material Testing Systems – Technical Description (Manual Part No. 100-146-666E)*. MTS Systems Corporation, May 2007.

[35] Patrick Larour, Benjamin Hackl, and Franz Leomann. Sensitivity analysis on the calculated bending angle in the instrumented bending test. June 2013.

[36] Patrick Larour, Benjamin Hackl, Franz Leomann, and Katarzyna Benedykt. Bending angle calculation in the instrumented three-point bending test. November 2012.

[37] Siemens AG. LOGO!24CE Logic Module Data Sheet (6ED1052-1CC08-0BA0). `https://docs.rs-online.com/eb10/0900766b8164240a.pdf`, March 2021.

[38] Diwaker Gupta and Amy Peterson. Statistical methods for materials testing. Technical Report MN/RC 2009-41, Industrial & Systems Engineering Program University of Minnesota 111 Church Street SE Minneapolis, MN 55455, December 2009.

[39] P.J. Blau. Lessons learned from the test-to-test variability of different types of wear data. *Wear*, 376-377:1830–1840, 2017. 21st International Conference on Wear of Materials.

[40] Elena Fiorese, Franco Bonollo, Giulio Timelli, Lars Arnberg, and Elisabetta Gariboldi. New classification of defects and imperfections for aluminum alloy castings. *International Journal of Metalcasting*, 9(1):55–66, January 2015.

[41] Hélder Puga. Casting and forming of advanced aluminum alloys. *Metals*, 10(4):494, April 2020.

[42] A. García Pérez, M. J. Gómez Silva, and A. De La Escalera Hueso. Automated defect recognition of castings defects using neural networks. *Journal of Nondestructive Evaluation*, 41(1), December 2021.

[43] Choong Do Lee. Effects of microporosity on tensile properties of A356 aluminum alloy. *Materials Science and Engineering A*, 464(1-2):249–254, February 2007.

[44] Jianli Zhao, Yilin Wang, Xiaowei Wang, and Yisheng Zhang. An experimental investigation of the material properties of the A356 aluminum alloy power fittings in the vacuum Die-Casting process. *Materials*, 17(6):1242, March 2024.

[45] Ewan Lordan, Jaime Lazaro-Nebreda, Yijie Zhang, Kun Dou, Paul Blake, and Zhongyun Fan. On the relationship between internal porosity and the tensile ductility of aluminium alloy die-castings. *Materials Science and Engineering A*, 778:139107, February 2020.

[46] Roemer J Janse, Tiny Hoekstra, Kitty J Jager, Carmine Zoccali, Giovanni Tripepi, Friedo W Dekker, and Merel Van Diepen. Conducting correlation analysis: important limitations and pitfalls. *Clinical Kidney Journal*, 14(11):2332–2337, May 2021.

[47] Avijit Hazra and Nithya Gogtay. Biostatistics series module 6: Correlation and linear regression. *Indian Journal of Dermatology*, 61(6):593, January 2016.

[48] Marie-Therese Puth, Markus Neuhäuser, and Graeme D. Ruxton. Effective use of Pearson's product–moment correlation coefficient. *Animal Behaviour*, 93:183–189, June 2014.

[49] J.R. Davis. *Tensile Testing*. ASM International, December 2004.

[50] Paweł Bogusz, Wiesław Krasoń, and Kamil Pazur. Application of digital image correlation for strain mapping of structural elements and materials. *Materials*, 17(11):2577, May 2024.

[51] Michał Marcin Bakalarz and Paweł Piotr Tworzewski. Application of digital image correlation to evaluate strain, stiffness and ductility of full-scale lvl beams strengthened by cfrp. *Materials*, 16(3), 2023.

[52] Akshat Agha. Effectiveness of 2D digital image correlation in capturing the fracture behavior of sheet metal alloys. *SAE International Journal of Materials and Manufacturing*, page 16, December 2022.

[53] Piab Group. Gripdon för gjutdelar – sprue/part grippers (produktnummer0213202). https://www.piab.com/sv-se/komponenterforrobotverktygeoat/formsprutningavplast/gripdonforgjutdelar/0213202, 2025.

Appendix A

Graphs and Images

A.1 Data for Section 2.13

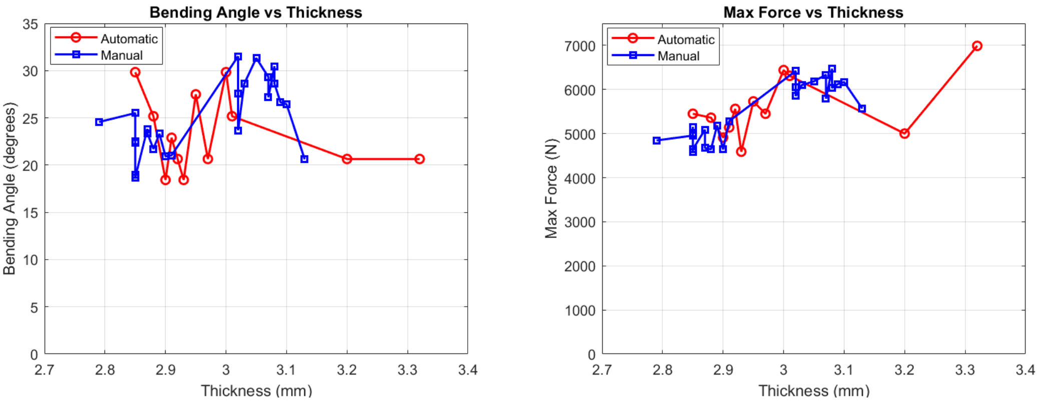

Figure A.1: Bending angle and peak force raw data, plotted with thickness

A.2 Test samples

Figure A.2: Aluminium test sample for bending (left) and tensile(right).

A.3 Robot platform and bending machine

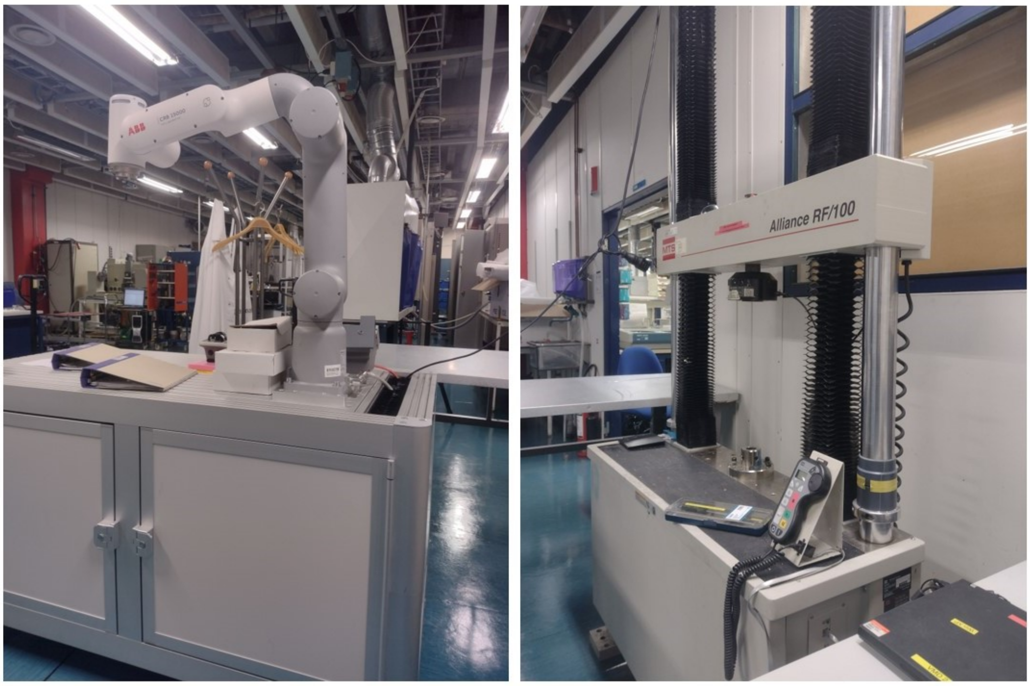

Figure A.3: Robot arm mounted on movable platform (left) and bending machine (right).

Appendix B

Calibration

B.1 Calibration for laser sensor

The calibration line is used to convert the value V, ranging from 0 to 1000 in the PLC memory, to a thickness measurement t_m in mm. Since only a difference in values is representative of a thickness, the line $y = mx + q$ is simplified to $y = mx$. The final relationship used is:

$$t_m = V/3.51 \tag{B.1}$$

(see Appendix C.2, Line 92).

B.2 Calibration of force and crosshead displacement

The calibration for force applied and crosshead displacement is done inside the bending machine software TestWorks 4 and can be customized.

$$F_{max} = 10 \cdot V_F, \tag{B.2}$$

$$S = V_S/3.33 \tag{B.3}$$

F_{max} is the peak force, S the crosshead displacement after preload, and V_F and V_S the corresponding values from 0 to 1000 in the PLC memory (see Appendix C.2, Lines 102 and 123).

Appendix C

Full code

C.1 PLC code

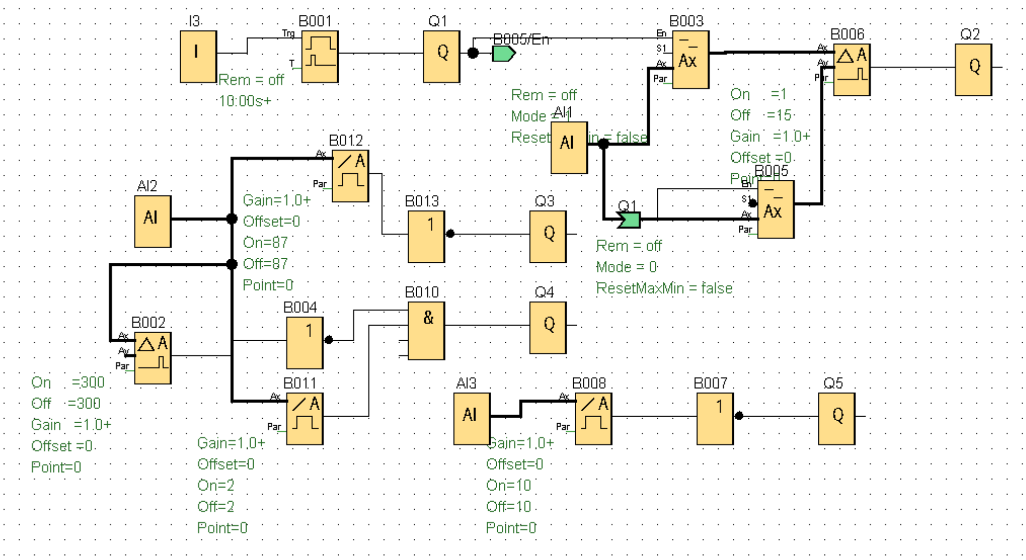

Figure C.1: FBD code for PLC

C.2 Python Code

```python
import cv2
import time
import openpyxl
import os
import pandas as pd
import math
#import snap7
#from snap7.util import set_bool
startTime=time.time()
count=0
# Get the path to the Desktop (works for both Windows and macOS/Linux)
```

```
desktop_path = os.path.join(os.path.expanduser("~"), "Desktop", "qr_codes.xlsx")
# Check if the Excel file already exists
if os.path.exists(desktop_path):
    # Open the existing Excel file
    wb = openpyxl.load_workbook(desktop_path)
    ws = wb.active # Get the active sheet
else:
    # Create a new workbook if the file doesn't exist
    wb = openpyxl.Workbook()
    ws = wb.active
    ws.title = "QR Codes"
PLC_IP= '192.168.0.1'
DB_Number= 1
Byte_index= 0
Bit_Index=0
#plc= snap7.client.Client()
#plc.connect(PLC_IP, 0, 1)
bending=True
istheFirst=0
# Open the camera (0 is usually the default webcam)
cap = cv2.VideoCapture(0)
# Create a QRCodeDetector object
qr_detector = cv2.QRCodeDetector()
preloaddisplacement= 591
while True:
    #this part is for adding values in the excel
    file_path = '\u202aC:\\Users\\S90\\Desktop\\LogFile.csv'
    cleaned_path = file_path.replace('\u202a', '')
    df = pd.read_csv(cleaned_path)
    # Capture frame-by-frame
    ret, frame = cap.read()
    if not ret:
        print("Failed to capture image")
        break
    # Detect and decode the QR code
    try:
        data, bbox, _ = qr_detector.detectAndDecode(frame)
        # If a QR code is detected and bbox is valid
        if data and bbox is not None:
            # Put the decoded data on the frame
            cv2.putText(frame, f'Data: {data}', (50, 50), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 0), 2)
            # Add a 10-second delay after detecting the QR code
            print("QR code detected! Creating new line...")
            istheFirst=istheFirst +1
            #data1=bytearray(plc.db_read(DB_Number, Byte_index, 1))
            #set_bool(data1, 0, Bit_Index, True)
            #plc.db_write(DB_Number, Byte_index, data1) #changes the value to 1 and activates Q7
            time.sleep(10) # Delay for 10 seconds
            last_row = df.iloc[-1] # -1 gives the last row
            if istheFirst==1:
                if last_row['I4']==1: #if the test type is confirmed to be BENDING
                    ws.append(["New batch - test data - 3 point bending"])
                    ws.append(["Sample number", "thickness ", "bending angle","max force ","slope"])
                elif last_row['I4']==0: #if testtype is confirmed to be tensile
                    ws.append(["New batch - test data - tensile"])
                    ws.append(["Sample number", "thickness [mm]","max force [N]"])
                #set_bool(data1, 0, Bit_Index, False)
            #plc.db_write(DB_Number, Byte_index, data1)
            # Append the QR code data to the Excel file
            ws.append([data]) # Add the data in a new row of the excel
            # Save the workbook to the desktop
            wb.save(desktop_path)
            print(f"QR code data saved: {data}")
    except Exception as e:
        print(f"Error detecting or decoding QR code: {e}")
    # Display the frame with QR code data
    cv2.imshow("QR Code Scanner", frame)
    last_row = df.iloc[-1] # -1 gives the last row
#execute this only if it's a bending test
    if last_row['I4']==1: #if the test type is confirmed to be BENDING
    #Thickness measurement
    #this are read from the csv file
        last_row = df.iloc[-1] # -1 gives the last row
        last_last_row = df.iloc[-2] #gives the row before the last one
        last_last_last_row= df.iloc[-3] #third row from the bottom
        diff= last_last_row['C4']-last_row['C4'] #this are the correct values for AI1, laser measurement
        #print(diff)
        if diff>4 and diff<10: #condition to log the difference
            #write the diff in the excel
            print("log new thickness measurement", diff)
            thickness=diff/3.51 #from the calibration, now is in mm
            last_row_excel = ws.max_row #gets the last row of the excel file
            ws.cell(row=last_row_excel, column=2, value=thickness)
```

```python
            wb.save(desktop_path)
            time.sleep(5)
    #logs the max force in excel
        diff4= last_row['AI3']-last_last_row['AI3']#this are the correct values for AI4, force applied
        diff5= last_last_row['AI3']- last_last_last_row['AI3']
        if diff4<0 and diff5>0:
            print("log new max force", last_last_row['AI3'])
            maxforce=last_last_row['AI3']*10 #converts the value in N
            last_row_excel = ws.max_row #gets the last row of the excel file
            ws.cell(row=last_row_excel , column=4, value=maxforce)
            wb.save(desktop_path)
        diff3= last_row['AI3']-last_last_row['AI3']#this are the correct values for AI4, force applied
        if diff3>0 and last_last_row['AI3']==0:
            preloaddisplacement=last_last_row['AI2'] #crossehad value (in 0-1000)at preload
            print("new displacement", preloaddisplacement)
        if (last_row['AI2']-last_last_row['AI2']) !=0:
            m=((last_row['AI3']-last_last_row['AI3'])*10)/((last_row['AI2']-last_last_row['AI2'])/3.333)
            if diff3>0 and last_last_row['AI3']<10 and m>0:
                print("new slope value:", m)
                last_row_excel = ws.max_row #gets the last row of the excel file
                ws.cell(row=last_row_excel , column=5, value=m)
                wb.save(desktop_path)
    #Bending angle measurement
    #this are read from the csv file
        diff2= last_row['AI3']-last_last_row['AI3']#this are the correct values for AI4, force applied
        if diff2<0 and last_row['AI3']<2: #by making the 2 smaller the measurement is more precise
            print("log new crosshead position", last_row['AI2'] )
            disp= last_row['AI2'] -preloaddisplacement #crosshead displacement form preload
            disp=disp/3.333 #now in mm
            r=0.4 #mm, punch radius
            D=30 #mm, roller diameter
            L=6.09 #mm, roller distance
            tm=thickness #sample thickness
            c=(D/2)+tm+r
            p= (D/2)+(L/2)
            W= math.sqrt(p**2+(disp-c)**2-c**2)
            angle= math.degrees(2*math.acos((W*p-c*(disp-c))/(p**2+(disp-c)**2)))
            last_row_excel = ws.max_row #gets the last row of the excel file
            ws.cell(row=last_row_excel , column=3, value=angle)
            wb.save(desktop_path)
#execute only if it's a tensile test
    elif last_row['I4']==0: #if testtype is confirmed to be tensile
    #this are read from the csv file
        last_row = df.iloc[-1] # -1 gives the last row
        last_last_row = df.iloc[-2] #gives the row before the last one
        last_last_last_row= df.iloc[-3] #third row from the bottom
        diff5= last_last_row['C4']-last_row['C4']
        if diff5<0 and last_row['C4']==0:
            startmeasure=time.time()
        if diff5>0 and last_row['C4']==0:
            endmeasure=time.time()
            thickness_=endmeasure-startmeasure #this is the time, needs to be converted to mm
            print("Logging new thickness value", thickness_)
            last_row_excel = ws.max_row #gets the last row of the excel file
            ws.cell(row=last_row_excel , column=3, value=thickness_)
            wb.save(desktop_path)
    # Break the loop when 'q' is pressed
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
#plc.disconnect()
# Release the camera and close all OpenCV windows
cap.release()
cv2.destroyAllWindows()
```

C.3 Robot code

```
MODULE MainModule
  CONST robtarget pickUp := [[993.12,93.50,136.26],[0.00184305,0.699293,0.714779,-0.00874605],[0,-1,1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];!cups
  CONST robtarget qrCode := [[949.20,-83.98,47.41],[0.00163261,0.699483,0.714591,-0.00892689],[-1,-1,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];!cups
    CONST robtarget before_qrCode := [[782.50,-83.94,47.35],[0.00158366,0.699486,0.714589,-0.0089546],[-1,-1,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];!cups
    CONST robtarget before_pickUp := [[969.36,93.38,162.60],[0.00181894,0.699326,0.714746,-0.00878559],[0,-1,1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];!cups
    CONST robtarget after_pickUp := [[841.00,93.28,162.49],[0.00175594,0.699364,0.714709,-0.00884688],[0,-1,1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];!cups
    VAR robtarget currentPos;
    CONST robtarget before_measurement:=
          [[729.36,-267.85,298.45],[0.00161964,0.699505,0.714571,-0.00890977],[-1,-1,0,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9
```

```
        E+09]];!cups
    CONST robtarget measurement:= [[748.22,-368.08,203.29],[0.113901,-0.670866,-0.668719,0.299633],[-1,-1,0,0],[9E+09,9E
        +09,9E+09,9E+09,9E+09,9E+09]];!cups
    CONST robtarget after_measurement := [[807.31,-314.79,110.04],[0.128322,-0.689958,-0.626499,0.339101],[-1,-1,0,0],[9E
        +09,9E+09,9E+09,9E+09,9E+09,9E+09]];!cups
    CONST robtarget after_testing:= [[-814.21,-103.97,81.26],[0.00509138,-0.711578,-0.701649,-0.0363391],[-2,-1,-1,0],[9E
        +09,9E+09,9E+09,9E+09,9E+09,9E+09]];!gripper
    CONST robtarget disposal_pickUp :=
        [[-1052.97,-113.96,17.09],[0.00502324,-0.706542,-0.70764,-0.00431698],[-2,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E
        +09,9E+09]];!gripper
    CONST robtarget disposal:= [[-86.18,-499.14,18.58],[0.0190478,-0.715392,0.698397,0.00969954],[-2,-1,1,0],[9E+09,9E
        +09,9E+09,9E+09,9E+09,9E+09]];!gripper
  PERS tooldata cups := [TRUE,[[0,250,70.5],[1,0,0,0]],[0.3,[0,0,45],[1,0,0,0],0.000076,0.00569,0.00577]];
    PERS tooldata gripper := [TRUE,[[ 0, -242,115.8],[1,0,0,0]],[0.3,[0,0,45],[1,0,0,0],0.000076,0.00569,0.00577]];
  CONST robtarget before_testing := [[-824.05,-107.80,126.28],[0.0363917,-0.703476,0.709768,0.00503275],[-2,-1,1,0],[9E
      +09,9E+09,9E+09,9E+09,9E+09,9E+09]];!cups
  CONST robtarget testing := [[-1026.92,-111.82,8.26],[0.00132636,0.779113,-0.626878,0.00233852],[-2,0,0,0],[9E+09,9E+09,9
      E+09,9E+09,9E+09,9E+09]];!cups
  CONST robtarget midwaytesting1 := [[633.51,-617.37,426.98],[0.00765945,0.850706,0.519315,-0.0809479],[-1,-1,0,0],[9E
      +09,9E+09,9E+09,9E+09,9E+09,9E+09]]; !cups
  CONST robtarget midwaytesting2 := [[-487.59,-738.07,426.94],[0.0457086,-0.986518,0.142026,0.067248],[-2,-1,0,0],[9E+09,9
      E+09,9E+09,9E+09,9E+09,9E+09]];!cups
  CONST robtarget return1 := [[-329.57,-509.56,545.56],[0.0190071,-0.715366,0.698424,0.00970655],[-2,-1,1,0],[9E+09,9E
      +09,9E+09,9E+09,9E+09,9E+09]];!tool0
  CONST robtarget return2 := [[300.53,-527.21,545.53],[0.0112175,-0.974053,0.225311,0.0181802],[-1,-1,1,0],[9E+09,9E+09,9E
      +09,9E+09,9E+09,9E+09]];!tool0
  CONST robtarget return3 := [[365.46,-394.59,545.53],[0.021308,-0.739627,-0.672679,0.00112997],[-1,-1,0,0],[9E+09,9E+09,9
      E+09,9E+09,9E+09,9E+09]]; !tool0
  CONST robtarget return4 := [[446.03,79.01,587.76],[0.0213177,-0.739603,-0.672705,0.00115511],[0,0,1,0],[9E+09,9E+09,9E
      +09,9E+09,9E+09,9E+09]]; !tool0
    PERS loaddata my_load:=[0.001,[0,0,0.001],[1,0,0,0],0,0,0];
    CONST robtarget after_qrCode := [[729.24,-91.73,228.01],[0.0017099,0.699398,0.714675,-0.00887841],[-1,-1,0,0],[9E+09,9
        E+09,9E+09,9E+09,9E+09,9E+09]];
    VAR num completedcycleCount_bending := 0;
    VAR num completedcycleCout_tensile :=0;
    VAR num n_first_stack:=12; !change to 0 if the user inputs it
    CONST robtarget pickUp2 := [[987.17,183.74,127.66],[0.00180553,0.699512,0.714563,-0.00891453],[0,-1,1,0],[9E+09,9E
        +09,9E+09,9E+09,9E+09,9E+09]];
    VAR robtarget disposal_error;
    CONST robtarget beforedisposal1 := [[-610.29,-125.48,154.84],[0.0221359,-0.975715,-0.216719,-0.0228719],[-2,-1,0,0],[9
        E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
    VAR num testtype:=0;
    CONST robtarget beforedisposal2 := [[-86.29,-474.07,149.95],[0.0189831,-0.715332,0.69846,0.0096874],[-2,-1,1,0],[9E
        +09,9E+09,9E+09,9E+09,9E+09,9E+09]];
    VAR robtarget pOffsetTarget;
    CONST robtarget tens_before_pickUp := [[861.46,26.96,126.66],[0.00420267,0.0314857,0.99943,-0.011459],[0,-1,0,0],[9E
        +09,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget tens_pickUp := [[870.14,88.35,42.13],[0.00960392,-0.0165421,0.999541,-0.0235006],[0,-1,0,0],[9E+09,9E
        +09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget tens_after_pickUp := [[390.28,129.12,400.92],[0.444508,-0.52413,-0.510271,0.517033],[0,0,1,0],[9E+09,9
        E+09,9E+09,9E+09,9E+09,9E+09]];
    PERS tooldata softGripper := [TRUE,[[0,0,300],[1,0,0,0]],[0.1,[0,0,45],[1,0,0,0],0.000076,0.00569,0.00577]];
    CONST robtarget tens_before_qrCode := [[842.16,-76.65,79.42],[0.0109854,0.717299,0.696658,-0.00529772],[-1,-1,0,0],[9E
        +09,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget tens_after_qrCode := [[662.54,-75.24,104.81],[0.421883,-0.512219,0.569136,-0.485521],[-1,-2,0,0],[9E
        +09,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget tens_qrCode := [[897.79,-76.68,47.80],[0.0109815,0.717302,0.696656,-0.00530085],[-1,-1,0,0],[9E+09,9E
        +09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget tensbeforemeasurement := [[902.84,-171.58,166.97],[0.32982,0.169322,0.477976,-0.796296],[0,-2,0,0],[9E
        +09,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget tensaftermeasurement := [[873.84,-171.67,166.91],[0.329766,0.169317,0.477997,-0.796307],[0,-2,1,0],[9E
        +09,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget tenshome := [[505.76,-76.38,572.84],[0.0241064,0.72101,0.690225,-0.0561474],[-1,-1,0,0],[9E+09,9E+09,9
        E+09,9E+09,9E+09,9E+09]];
    CONST robtarget tens_midwaytesting1 := [[61.28,-693.78,528.06],[0.0958796,-0.0601222,0.720197,-0.684477],[-1,1,1,0],[9
        E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget tens_midwaytesting2 := [[-636.49,-282.80,528.07],[0.319913,-0.369766,-0.620939,0.612669],[-2,1,1,0],[9
        E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget tens_testing := [[-1089.38,-189.56,96.20],[0.330586,-0.37914,-0.600713,0.621378],[-2,1,1,0],[9E+09,9E
        +09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget tens_beforeTesting := [[-1017.51,-315.88,96.18],[0.330597,-0.379166,-0.600709,0.62136],[-2,1,1,0],[9E
        +09,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget tens_duringTesting := [[-943.35,-118.66,96.13],[0.389514,-0.447151,-0.560054,0.57851],[-2,1,1,0],[9E
        +09,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget tens_disposal_upper :=
        [[-1067.91,-183.78,139.27],[0.368677,-0.354257,-0.621869,0.593176],[-2,1,1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E
        +09]];
    CONST robtarget tens_disposal_lower := [[-1061.65,-173.36,86.28],[0.33724,-0.411471,-0.598387,0.599077],[-2,1,1,0],[9E
        +09,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget tens_disposal_general :=
        [[-269.14,-129.66,52.68],[0.0175953,0.0412698,0.998137,-0.0413612],[-2,0,2,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E
        +09]];
    CONST robtarget tens_disposal_upper_afterpickup :=
```

```
[[-719.33,142.16,80.80],[0.364359,-0.365372,-0.621605,0.589366],[1,1,1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget tens_disposal_lower_afterpickup := [[-719.33,142.16,80.80],[0.364359,-0.365372,-0.621605,0.589366],[1,1,1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
    !CONST robtarget tens_midwaydisposal := [[-417.31,-178.46,356.09],[0.0685055,-0.211178,0.755584,-0.616283],[1,0,1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget tens_midwaydisposal := [[-718.61,-247.19,322.68],[0.337264,-0.411542,-0.598363,0.599039],[-2,1,1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
    CONST robtarget tens_midwaydisposal2 := [[-805.92,-208.46,159.85],[0.341957,-0.36695,-0.643607,0.578086],[-2,1,1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
PROC main()
    reset_all_scalableDO;
    RESET ABB_Scalable_IO_0_DO16; !has to reset testType from the previous cycle
    IF completedcycleCount_bending=0 AND completedcycleCout_tensile=0 THEN
        TPErase;
        TPReadFK testtype, "What type of test will be executed? ", "Bending", "Tensile", "trial", stEmpty, stEmpty;
    ENDIF
    IF testtype = 1 THEN
        SET ABB_Scalable_IO_0_DO16; !this is the verification for bending test
        pickUp_procedure_bending;
        qrCode_procedure_bending;
        measurement_procedure_bending;
        testing_procedure_bending;
        disposal_procedure_bending;
        return_procedure_bending;
    ELSEIF testtype = 2 THEN
        pickUp_procedure_tensile;
        qrCode_procedure_tensile;
        measurement_procedure_tensile;
        testing_procedure_tensile;
        disposal_procedure_tensile;
    ELSEIF testtype= 3 THEN
        MoveJ tens_before_pickUp, v50, z0, softGripper; !goes to home position
        WaitTime 1;
        currentPos := CRobT(); ! Get current position
        WHILE ABB_Scalable_IO_0_DI6 = 0 DO !until laser detect correct position on the z
            MoveL Offs(currentPos, -5, -5, 0), v50, z0, softGripper; ! Move in steps of 5 mm
            currentPos := CRobT(); ! Update current position
        ENDWHILE
        WaitTime 1;
    ELSE
        ErrRaise "Custom error", 90001, "error", "error","error", "error", "error";
    ENDIF
    ERROR !deals with errors in the entire code
        TPWrite "An error occured, trying reset...";
        IF ABB_Scalable_IO_0_DI1 =1 THEN !detects if the vacuum is on(there's a sample attached)
            MoveJ disposal_error, v50, z0, cups;
            RESET ABB_Scalable_IO_0_DO2; !deactivates vacuum
            SET ABB_Scalable_IO_0_DO3; !activates and deactivates blow off
            WaitTime 1;
            RESET ABB_Scalable_IO_0_DO3;
        ENDIF
        MoveJ wi_homePosition, v50, z0, tool0; !goes to home position
        StartMoveRetry; !restats the program
ENDPROC
PROC pickUp_procedure_bending()
        MoveL before_pickUp, v50, z0, cups;
        IF completedcycleCount_bending=0 THEN
            !TPReadNum n_first_stack, "How many samples are there in the first stack?";
        ENDIF
        IF completedcycleCount_bending <n_first_stack THEN !picks up only the correct number of samples from the first stack
            MoveL pickUp, v20, z0, cups;
        ELSE
            MoveL pickUp2, v20, z0, cups;
        ENDIF
        SET ABB_Scalable_IO_0_DO2; !activates vacuum
        currentPos := CRobT(); ! Get current position
        WHILE ABB_Scalable_IO_0_DI1 = 0 DO !until vacuum is detected, goes down
            MoveL Offs(currentPos, 0, 0, -10), v20, z0, cups; ! Move down in steps of 10 mm
            currentPos := CRobT(); ! Update current position
        ENDWHILE
        IF completedcycleCount_bending <n_first_stack THEN
            MoveL pickUp, v20,z0,cups; !goes back up straight, to not move the samples below
        ELSE
            MoveL pickUp2, v20, z0, cups;
        ENDIF
        MoveL after_pickUp,v100,z0, cups;
ENDPROC
PROC qrCode_procedure_bending()
        MoveL before_qrCode, v200, z0, cups;
```

```
        MoveL qrCode, v200, z0, cups;
        WaitTime 7; !reads QR-code
        MoveL before_qrCode, v200, z0, cups;
        MoveL after_qrCode, v200, z0, cups;
ENDPROC
PROC measurement_procedure_bending()
    MoveL before_measurement,v200,z0,cups;
    MoveL measurement, v100, z0, cups;
    SET ABB_Scalable_IO_0_DO5; ! this is the output doMeasurement seen in the PLC as doMeasure, I1
    WaitTime 5;
    RESET ABB_Scalable_IO_0_DO2; !deactivates vacuum
    SET ABB_Scalable_IO_0_DO3; !activates and deactivates blow off
    WaitTime 1;
    RESET ABB_Scalable_IO_0_DO3;
    WaitTime 5; !waits for the measurement to take place until the end
    !WaitDI ABB_Scalable_IO_0_DI3, 1, \MaxTime:=30; !waits the input continueTest_measurement from the PLC, maxtime 1 min, after calls the error handler !deactivated
    MoveL after_measurement, v50, z0, cups;! goes to pick up the sample after measurement
    FCRefForce \Fz:=-5; ! Setup the force reference with 5N in Z-direction of the world frame
    FCAct cups; ! Activate Force Control
    FCRefStart;! Start moving the robot to achieve the specified force
    WaitTime 10;
    SET ABB_Scalable_IO_0_DO2; !activates vacuum
    FCRefStop; ! Stop the reference values
    FCDeact; ! Deactivate force control
    MoveL before_measurement, v100, z0, cups;
    RESET ABB_Scalable_IO_0_DO5; !resets output doMeasurement to be ready for next cycle
    RESET ABB_Scalable_IO_0_DO5; !deactivates the plc circuit to do the measurement
ENDPROC
PROC testing_procedure_bending()
    MoveJ midwaytesting1, v1000, z0, cups;
    MoveJ midwaytesting2, v1000, z0, cups;
    MoveJ before_testing, v1000,z0, cups; !movest into the testing area
    MoveL testing, v50,z0, cups;
    WaitTime 3;
    RESET ABB_Scalable_IO_0_DO2; !deactivates vacuum
    SET ABB_Scalable_IO_0_DO3; !activates and deactivates blow off
    WaitTime 1;
    RESET ABB_Scalable_IO_0_DO3;
    MoveL testing, v50,z0, cups;
    MoveL before_testing, v50, z0, cups; !gets out of the way
    WaitDI ABB_Scalable_IO_0_DI4, 1,\MaxTime:=120; !positon check
    TPWrite "Position check passed, test starts in 5 seconds";
    WaitTime 5;
    SET ABB_Scalable_IO_0_DO12; !this is start test
    MoveJ after_testing, v100, z0, gripper; !turns the tool around
    RESET ABB_Scalable_IO_0_DO12; !this needed to be reset
ENDPROC
PROC disposal_procedure_bending()
    WaitDI ABB_Scalable_IO_0_DI7, 1, \MaxTime:=300; !waits for the test to be finished
    MoveL disposal_pickUp, v100, z0, gripper;
    WaitTime 2;
    SET ABB_Scalable_IO_0_DO6; !closes the gripper
    MoveL after_testing, v100, z0, gripper;
    MoveL beforedisposal1, v100, z0, gripper;
    MoveL beforedisposal2, v100, z0, gripper;
    IF completedcycleCount_bending<6 THEN
        pOffsetTarget := disposal; ! Copy original position
        pOffsetTarget.trans := disposal.trans + [80*completedcycleCount_bending, 0, 0]; ! Add 60mm in X per completed cycle
        MoveL pOffsetTarget, v50, z0, gripper;
        RESET ABB_Scalable_IO_0_DO6; !drops tested sample
    ELSEIF completedcycleCount_bending>5 AND completedcycleCount_bending<12 THEN
        pOffsetTarget := disposal; ! Copy original position
        pOffsetTarget.trans := disposal.trans + [80*(completedcycleCount_bending-6), 40, 0]; ! Add 60mm in X per completed cycle
        MoveL pOffsetTarget, v50, z0, gripper;
        RESET ABB_Scalable_IO_0_DO6; !drops tested sample
    ELSE
        pOffsetTarget := disposal; ! Copy original position
        pOffsetTarget.trans := disposal.trans + [80*(completedcycleCount_bending-12), 80, 0]; ! Add 60mm in X per completed cycle
        MoveL pOffsetTarget, v50, z0, gripper;
        RESET ABB_Scalable_IO_0_DO6; !drops tested sample
    ENDIF
ENDPROC
PROC return_procedure_bending()
    MoveL return1, v300, z0, tool0;
    MoveJ return2, v300, z0, tool0;
    MoveJ return3, v300, z0, tool0;
    MoveJ return4, v300, z0, tool0;
    completedcycleCount_bending := completedcycleCount_bending + 1; !counts the number of cycles
ENDPROC
PROC reset_all_scalableDO()
```

```
    RESET ABB_Scalable_IO_0_DO1;
    RESET ABB_Scalable_IO_0_DO2;
    RESET ABB_Scalable_IO_0_DO3;
    RESET ABB_Scalable_IO_0_DO4;
    RESET ABB_Scalable_IO_0_DO5;
    RESET ABB_Scalable_IO_0_DO6;
    RESET ABB_Scalable_IO_0_DO7;
    RESET ABB_Scalable_IO_0_DO8;
    RESET ABB_Scalable_IO_0_DO9;
    RESET ABB_Scalable_IO_0_DO10;
    RESET ABB_Scalable_IO_0_DO11;
    RESET ABB_Scalable_IO_0_DO12;
    RESET ABB_Scalable_IO_0_DO13;
    RESET ABB_Scalable_IO_0_DO14;
    RESET ABB_Scalable_IO_0_DO15;
    RESET ABB_Scalable_IO_0_DO16;
ENDPROC
PROC pickUp_procedure_tensile()
    MoveL tens_before_pickUp, v100, z0, softGripper;
    WaitTime 1;
    currentPos := CRobT(); ! Get current position
        WHILE ABB_Scalable_IO_0_DI6 = 0 DO !until laser detect correct position, goes down
            MoveL Offs(currentPos, 0, 0, -5), v10, z0, softGripper; ! Move down in steps of 5 mm
            currentPos := CRobT(); ! Update current position
        ENDWHILE
    WaitTime 1;
    SET ABB_Scalable_IO_0_DO15; !closes the soft gripper
    MoveL tens_before_pickUp, v100, z0, softGripper;
ENDPROC
PROC qrCode_procedure_tensile()
    MoveJ tens_before_qrCode, v100, z0, softGripper;
    MoveL tens_qrCode, v100, z0, softGripper;
    WaitTime 7; !wait for QR-code detection
    Movel tens_before_qrCode, v100, z0, softGripper;
ENDPROC
PROC measurement_procedure_tensile()
    MoveJ tensbeforemeasurement, v20, z0, softGripper;
    MoveL tensaftermeasurement, v5, z0, softGripper; !does the measurement, recod the time
    MoveJ tenshome, v100, z0, softGripper;
ENDPROC
PROC testing_procedure_tensile()
    MoveJ tens_midwaytesting1, v300, z0, softGripper;
    MoveJ tens_midwaytesting2, v300, z0, softGripper;
    MoveJ tens_beforeTesting, v300, z0, softGripper; !right above the machine lower gripper
    MoveL tens_testing, v30, z0, softGripper;
    SET ABB_Scalable_IO_0_DO9; !this is close the lower gripper on the machine
    WaitTime 3; !closes for 3 seconds
    RESET ABB_Scalable_IO_0_DO9; !the gripper finished the closing procedure
    WaitTime 2;
    RESET ABB_Scalable_IO_0_DO15; !opens the soft gripper
    MoveL tens_duringTesting, v50, z0, softGripper;
    WaitTime 1;
    SET ABB_Scalable_IO_0_DO10; !this is start test on the machine (with the finger)
    WaitTime 3;
    RESET ABB_Scalable_IO_0_DO10;
    WaitTime 15; !waits
    SET ABB_Scalable_IO_0_DO13; !close the upper gripper
    WaitTime 3;
    RESET ABB_Scalable_IO_0_DO13;
    TPWrite "Test starting...";
    !execute test
ENDPROC
PROC disposal_procedure_tensile()
    WaitDI ABB_Scalable_IO_0_DI9, 1; !test finished and grippers in position
    MoveL tens_disposal_lower, v50, z0, softGripper;
    WaitTime 1;
    currentPos := CRobT(); ! Update current position
    WHILE ABB_Scalable_IO_0_DI6 = 0 DO !until laser detect correct position
         MoveL Offs(currentPos, -10, -5, 0), v10, z0, softGripper; ! Move in steps of 5 mm
         currentPos := CRobT(); ! Update current position
    ENDWHILE
    WaitTime 1;
    SET ABB_Scalable_IO_0_DO15; !closes the soft gripper
    WaitTime 1;
    SET ABB_Scalable_IO_0_DO11; !opens the lower gripper
    WaitTime 3;
    RESET ABB_Scalable_IO_0_DO11;
    MoveL tens_beforeTesting, v20, z0, softGripper; !this point is on the side
    MoveJ tens_midwaydisposal, v100, z0, softGripper;
    MoveJ tens_disposal_general, v100, z0, softGripper;
    WaitTime 1;
    RESET ABB_Scalable_IO_0_DO15; !opens the soft gripper
    WaitTime 1;
```

```
    MoveL tens_midwaydisposal, v100, z0, softGripper;
    MoveJ tens_midwaydisposal2, v100, z0, softGripper;
    MoveL tens_disposal_upper, v50, z0, softGripper;
    WaitTime 1;
    currentPos := CRobT(); ! Get current position
        WHILE ABB_Scalable_IO_0_DI6 = 0 DO !until laser detect correct position
            MoveL Offs(currentPos, -10, -5, 0), v10, z0, softGripper; ! Move in steps of 5 mm
            currentPos := CRobT(); ! Update current position
        ENDWHILE
    WaitTime 1;
    SET ABB_Scalable_IO_0_DO15; !closes the soft gripper
    WaitTime 1;
    SET ABB_Scalable_IO_0_DO14; !opens the upper gripper
    WaitTime 3;
    RESET ABB_Scalable_IO_0_DO14; !finish opeining the upper gripper
    MoveL tens_beforeTesting, v20, z0, softGripper; !this point is on the side
    MoveL tens_midwaydisposal, v100, z0, softGripper;
    MoveJ tens_disposal_general, v100, z0, softGripper;
    WaitTime 1;
    RESET ABB_Scalable_IO_0_DO15; !opens the soft gripper, drop sample
    WaitTime 1;
    MoveJ tens_midwaytesting2, v300, z0, softGripper;
    MoveJ tens_midwaytesting1, v300, z0, softGripper;
    MoveJ tenshome, v300, z0, tool0;
    completedcycleCout_tensile:=completedcycleCout_tensile+1; !counts the number of cycles of tensile
ENDPROC
ENDMODULE
```