Conveyor Belt Sorting System

Introduction

The project aims to design an innovative conveyor belt sorting system utilizing a microcontroller for the automated detection and sorting of objects by size and weight. The system will use an IR sensor to determine the size of the object based on the time it obscures the light, relative to belt speed, and using a force sensing resistor to detect of the object is above the weight threshold. An actuator system, using a micro servo, will sort objects to their respective stations, with an LCD display showing the total objects, and number of objects in stations A and B. The LCD also presents the weight of the object and the size.

Design Specification

System Purpose and Overview
The Conveyor Belt Sorting System utilizes sensors to detect the size and weight of objects, directing them to different stations. It includes an actuator system to control the direction of the object on the conveyor belt, allowing them to be delivered to designated stations. The system is also able to count objects and a user interface for monitoring/ control is available. The aim of the system is to automate the sorting process.

Components
Conveyor Belt - Provides the transportation mechanism for objects.
IR LEDs - Allows interruption detection to measure the size of objects and will count the number of objects directed to
Strain gauge (FSR) - Allows the weight of objects to be measuring.
Actuator System - Includes actuator motor and arm to control the direction of the objects to different stations.
LCD display and button - Provides User Interface for monitoring/ control of the entire system.
MCU – Required for system control and decision-making based on sensor inputs.

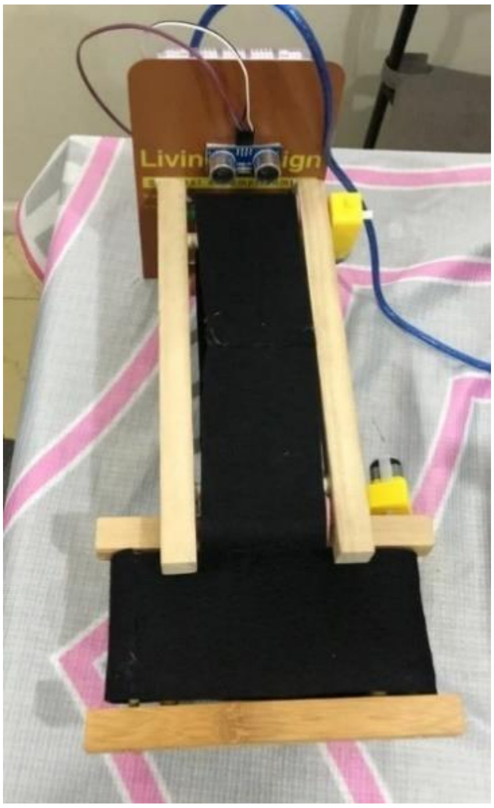
Operation
Object is placed atop of FSR on the conveyor belt (idle at start).
FSR measures weight of the object.
Operator uses control on button to start the conveyor belt.
As object move along the belt, the IR Led measures its size through interruption detection.
Based on the size and or weight of the object, the actuator system adjusts the direction of the arm to direct objects to one of 2 designated stations.
An IR LED at each station connects to LCD which keeps track of the number of objects at each station.
The LCD system allows for functions such as switching from weight to size sorting (or vice versa) and viewing the count of objects at each station.
We use a button instead of a keypad because it is simpler and requires less troubleshooting. The button system allows for functions such as emergency stops, pausing, and overriding destinations.

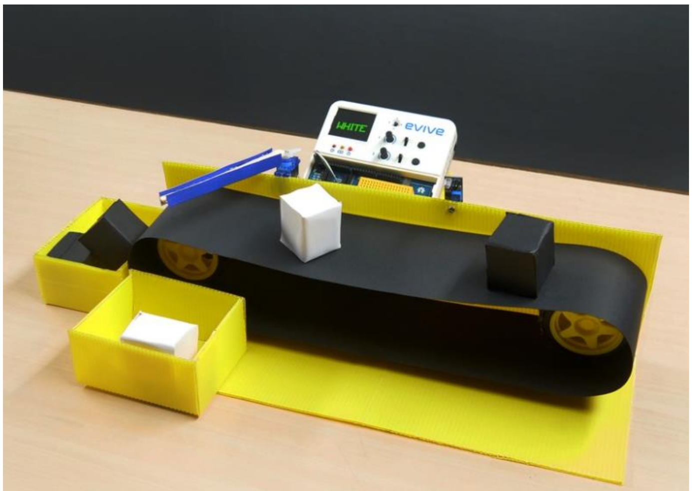
Safety Features
Emergency stop button - Immediately stops the operation of the conveyor belt in case of emergencies.
Safety guards - Physical barriers to prevent access to moving parts of the system or objects falling of the belt.
Overload protection - Ensures the system shuts down if it exceeds predefined weight or size limits and protects MCU.

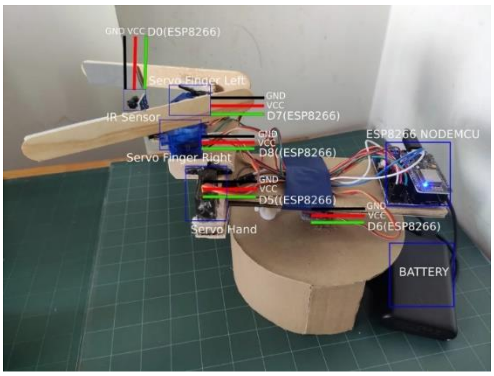
Accessibility and Usability
Cable management: Allow for easier troubleshooting and rerouting.
Colour coded/ labelled wires: Decreases human error.
Enclosure Design: Allows for ease of transport and provides rigidity (weight of object wont impact system operation).

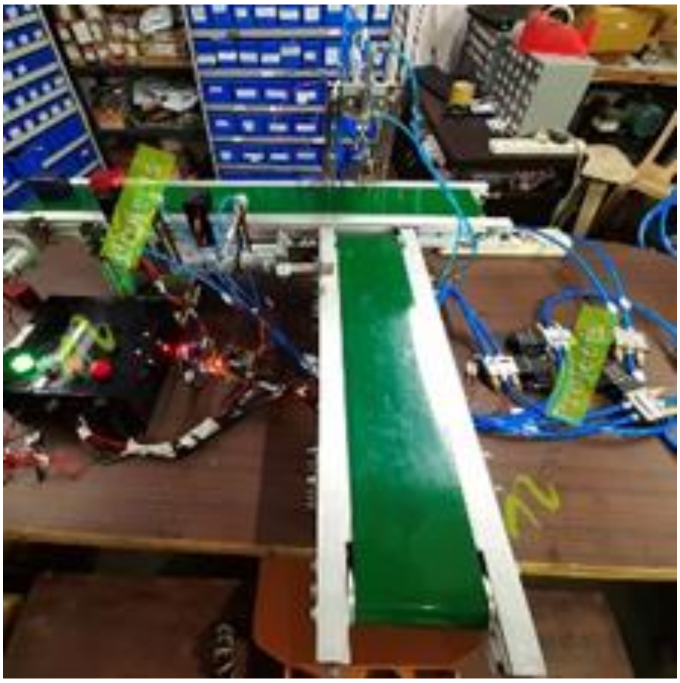
Review of the Literature on Current Systems Design Concepts

Conveyor belt systems are essential for industrial automation, enabling efficient object movement. The integration of Force Sensitive Resistor (FSR) technology provides precise object detection and size measurement. The PIC16F887 microcontroller, with a 16x2 LCD display, enhances user interaction by providing real-time feedback on motion detection. Researching existing literature review highlighted how FSR integration, transforms conveyor belt systems, offering efficient object movement, accurate sorting, and user-friendly interfaces for modern manufacturing processes. Our collaborative research on existing systems provided ideas and inspiration for our conveyor belt project design. This investigation provided insights into various configurations and functionalities that our conveyor belt system could embody. Some example systems we researched are provided below.



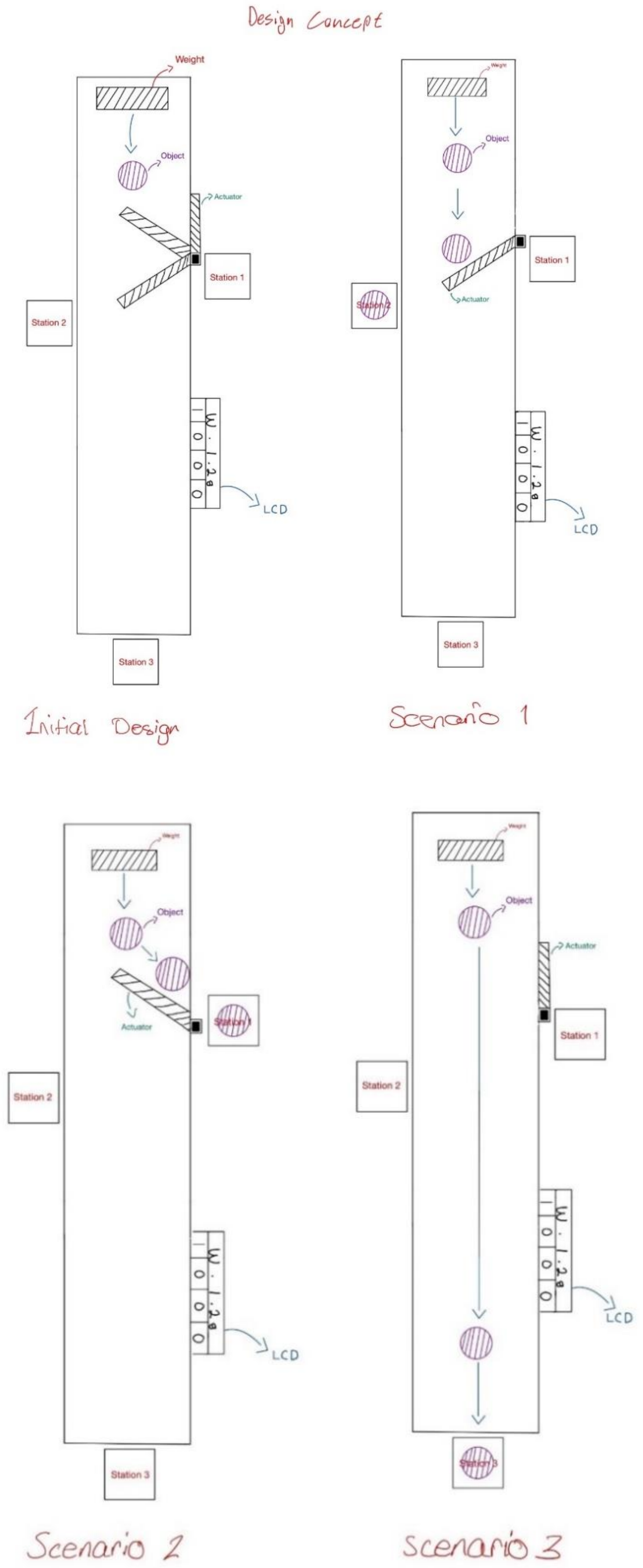




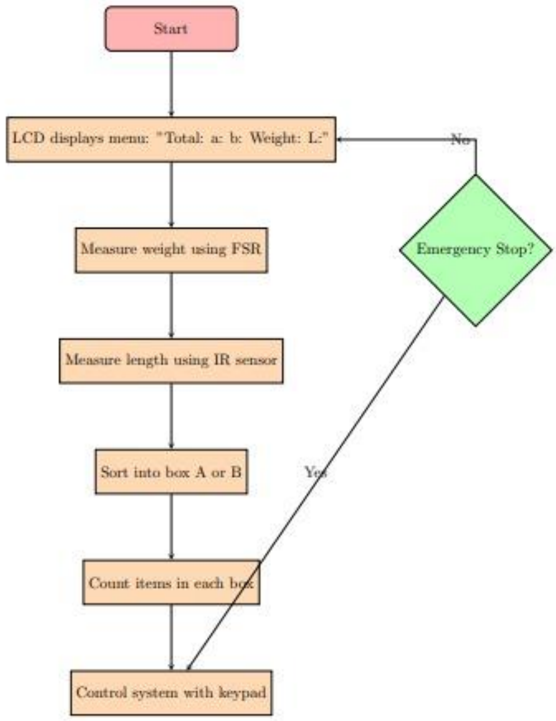


Initial Design Concept

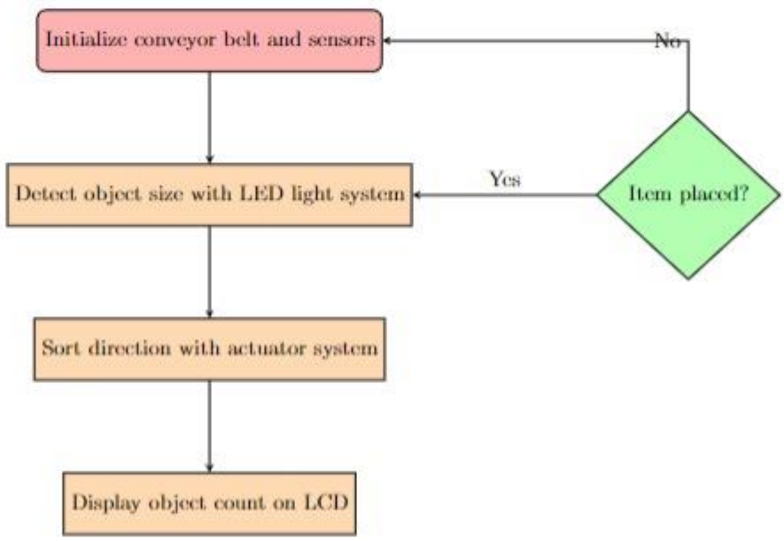
An initial concept for the conveyor belt system and crafted three detailed scenarios to accompany it. These scenarios serve as practical examples demonstrating the intended functionality of the project upon completion.



Initial Design Flowchart



Focused on sorting items based on weight categories



System is ready to detect and sort objects

Design (Multisim)

Power supply and LEDs
There are two LEDs in the schematic. LED1 is connected to a 300Ω resistor and a 5V supply (V1). LED2 is connected in series with a 15Ω resistor and another 5V supply (V3). It seems to be controlled by a force-sensitive resistor (FSR), which is a type of resistor that changes its resistance with force or pressure. The FSR and a 10kΩ resistor form a voltage divider, affecting the voltage at V3 and hence the brightness of LED2.

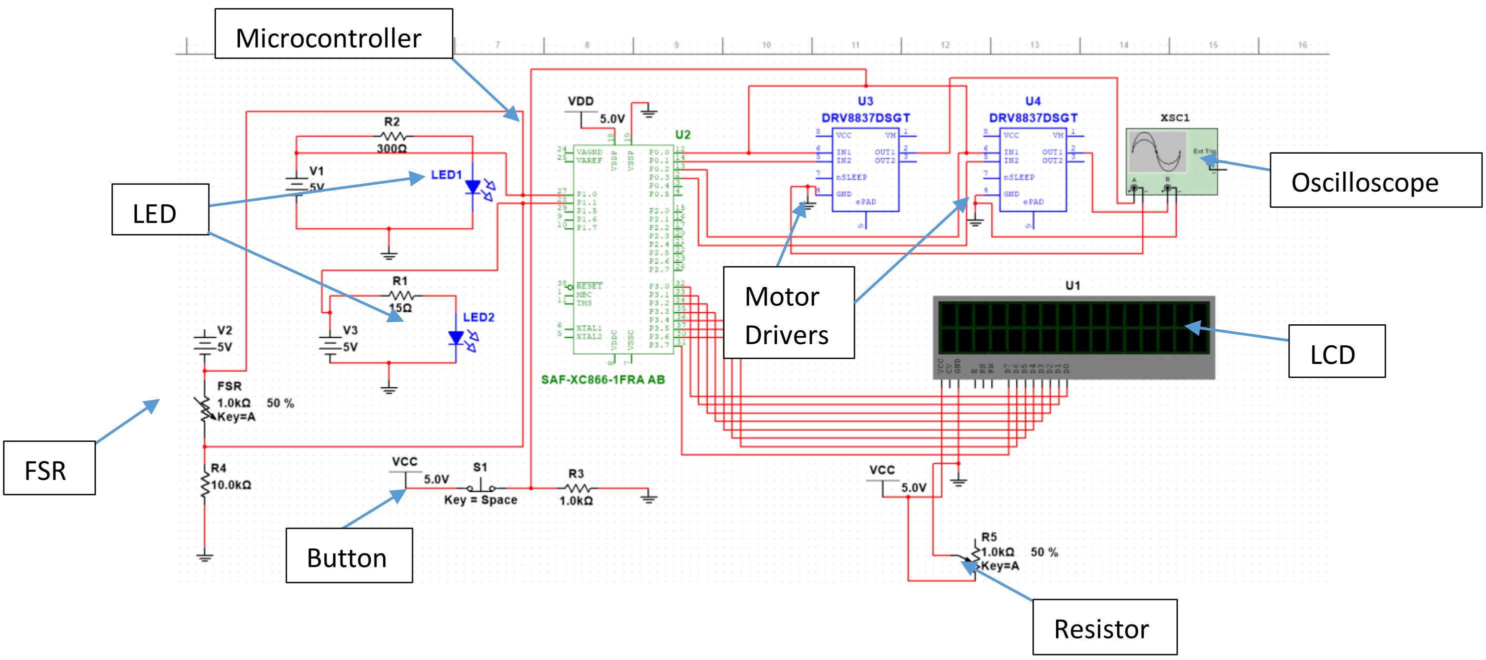
Microcontroller
U2 is a microcontroller, labeled 'SAF-XC866-1FRA AB'. This is the chip of the circuit, which would be programmed to perform various tasks, including controlling the LEDs, receiving sensor inputs, and driving motors via the motor drivers (U3 and U4).

Motor Drivers
U3 and U4 are labeled 'DRV8837DSGT', which are motor driver ICs. They're responsible for controlling motors, and this system can control two motors independently.

External Oscilloscope Trigger
There's an external oscilloscope trigger (XSC1), which seems to be used for monitoring or debugging purposes.

User Interface Elements
S1, which is labeled 'Key = Space', indicates a button that serves as a user input device. Pressing this button may trigger an event or interruption in the microcontroller. R5 is labeled '1.0kΩ 50% Key=A'.

LCD Display
U1 is an LCD display, connected to the microcontroller via multiple data lines.



Design

Our project faced simulation challenges in Multisim, leading us to modify our circuit design to use dual power sources for the LEDs and work around the absence of the PIC16f18877 in the student library. We developed a conveyor belt system equipping it with sensors such as FSR and an optical IR LED pair for detecting objects. Movement is driven by a Bi-Polar stepper motor using an L298N H-Bridge for precise control, while an LCD screen and LEDs provide user interaction, along with an emergency switch. Using MPLAB X IDE for C programming helped in organizing and scaling our code, supported by MPLAB's simulation tools for reliable operation. The system measures an object's weight and width to sort it into the correct station and uses the PIC and an H-bridge for directional control of the belt.

Implementation and Testing

LCD and FSR Flowcode simulation

During this project, we faced issues with our LCD code, prompting us to use Flowcode for simulation. The original code malfunctioned, leading us to replace the FSR with a push-button in the simulation for more accurate item placement. This adjustment, allowing for intentional activation, simplified the process and enhanced understanding. We programmed a menu that prompts users to place an item on the system, which then counts the item once placed, mimicking real conveyor system operations.

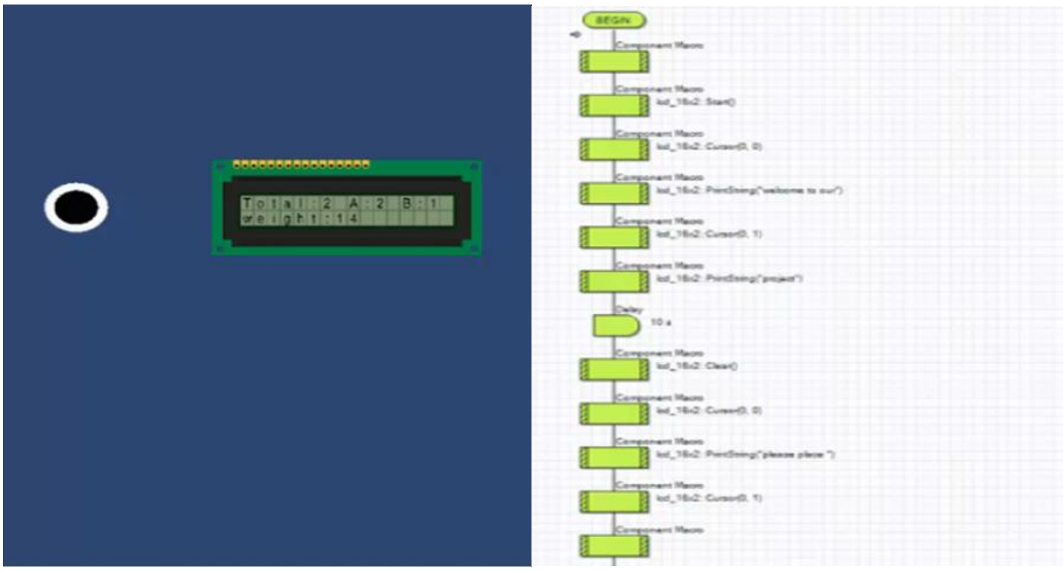


Figure 1 - Flow code Simulation

H Bridge Stepper Motor Circuit

The H-Bridge circuit is used to control the direction that the stepper motor rotates. It allows voltage to be applied across a load in either direction; four switches, diodes for back EMF protection, and a control circuit built within a microcontroller are necessary parts. It takes in 4 input values from the FSR and outputs to 4 pins for the stepper motor.

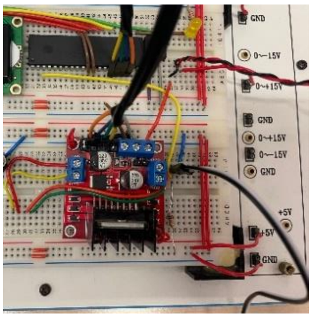


Figure 2 - H bridge

Servo motor circuit

The PIC16F1937 was chosen for its ability to generate a PWM signal, as this task was found to be more complex with the PIC16F18877. The setup involves adjusting the following registers: CCP1CON to set the PWM mode, T2CON to select the timer and PR2 to determine the PWM period. Adjusting the duty cycle, which dictates the servo motor's position, requires modifications to the CCPR1L register and the CCP1CON register. Additionally, the I/O pin connected to the CCP1 output is configured as an output pin.

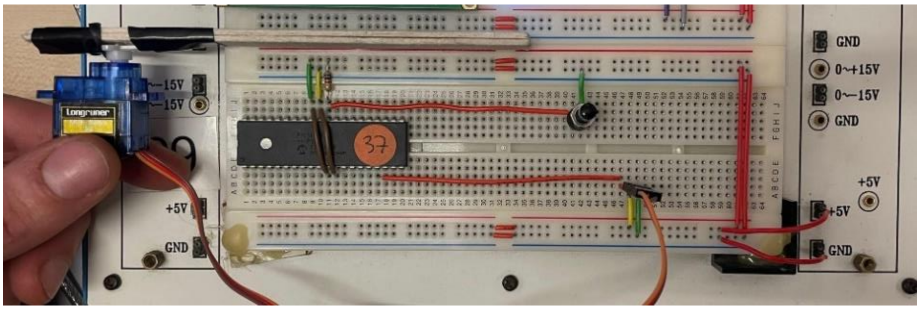


Figure 3 - Micro Servo

LCD circuit

Below is the breadboard setup for the 16x2 LCD display, which serves to display important outputs from our systems. This display is connected to a microcontroller on the breadboard, functioning as an interface for data presentation. The LCD features LED backlighting, enhancing readability under different lighting conditions through a blue-hued illumination. The microcontroller communicates with the LCD display, allowing it to act as an output device that presents data visually on the screen.

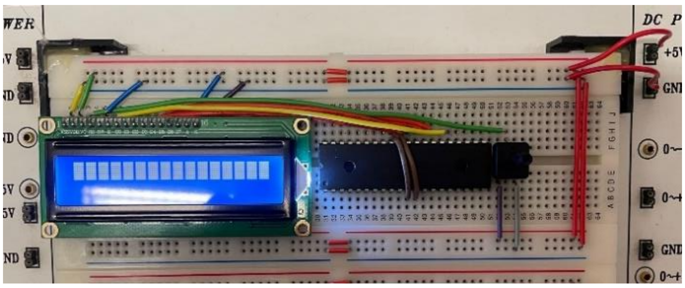
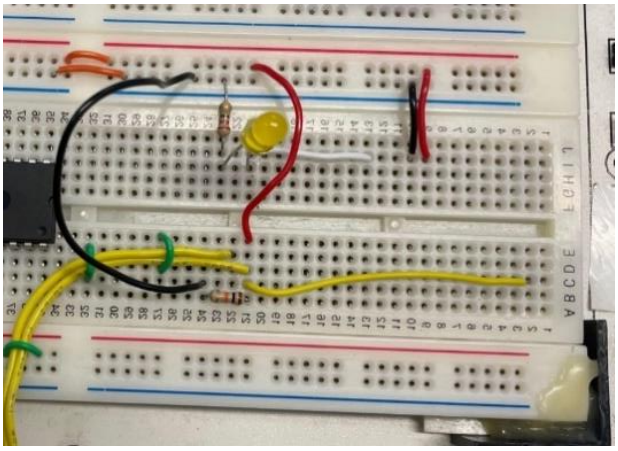


Figure 4 - LCD

FSR circuit

The FSR (Force Sensing Resistor) is connected in series with a fixed reistor of 10k ohms forming a voltage divider which is connected in to the ADC (analouge to digital converter) pin of the MCU which measures the voltage it recieves and can output a signal to any of the I/O pins, this is was first tested using an LED but later connected to the servo motor controlling its position according to the weight of the object.

Figure 5 - FSR



IR LED Circuit

This circuit uses the photodiode as the emitter and the phototransistor as the receiver. The phototransistor requires a control voltage and, just like a transistor, allows for more voltage and current flow when it receives a signal from the photodiode. A program was then written to input the output voltage into the PIC and detect whether it is high or low, subsequently measuring the time. To test if the program is working, an output LED was used; it was programmed to turn on when the PIC is counting.

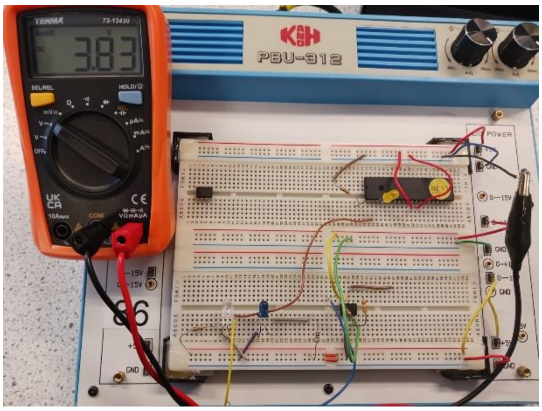


Figure 6 - IR circuit

Final design

Below is an image of the fully assembled conveyor belt sorting system, with each component clearly labelled:

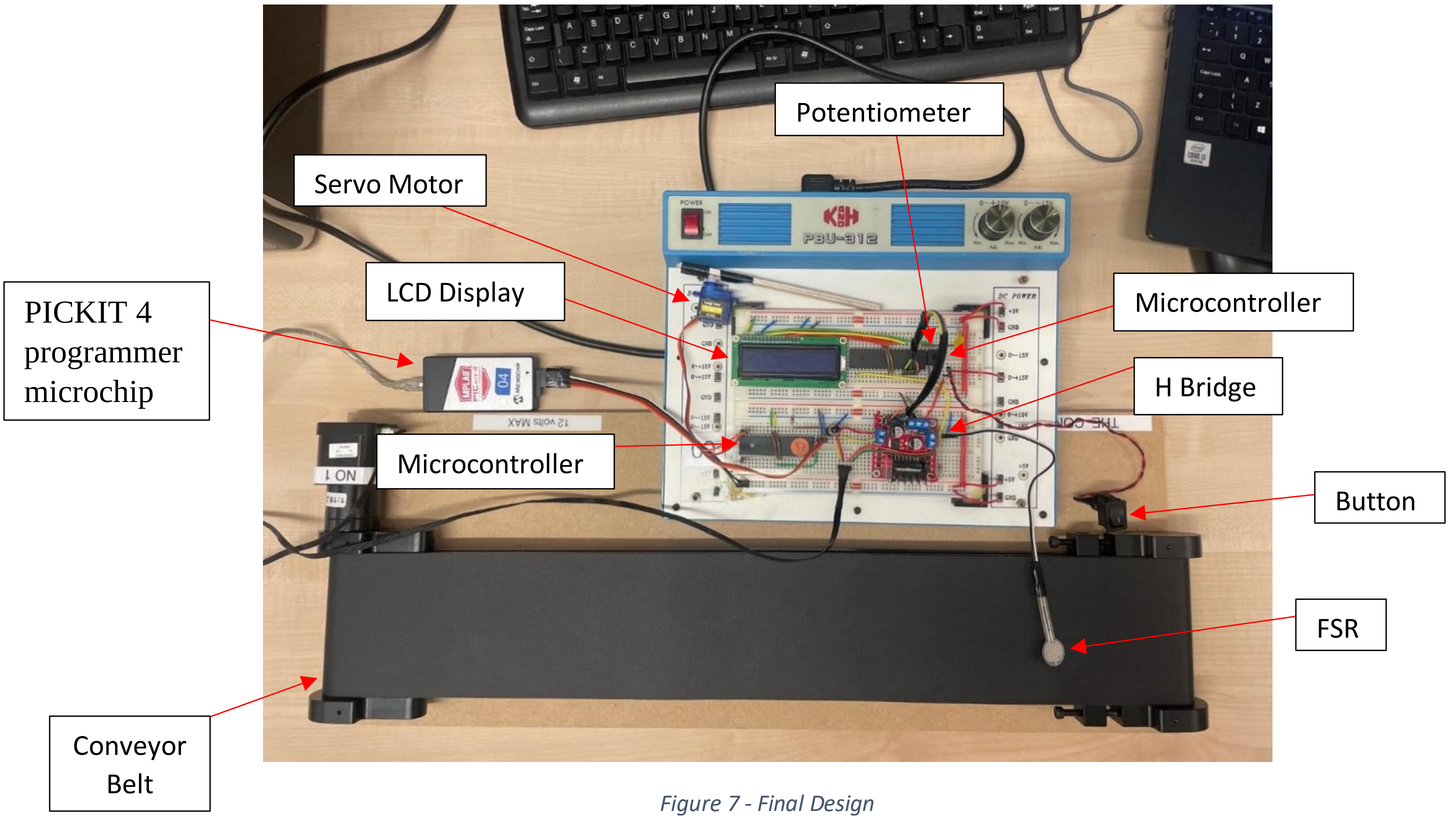


Figure 7 - Final Design

Comparison

Comparing the existing systems with our final design, we observed differences such as belts use callipers to sort the items, ours used the servo motors to push the items. Furthermore, the other projects use two conveyor belts, ours used one reducing the complexity. Comparing our initial to the final design, we made a few changes such as reducing the number of stations due to the FSR being unable to detect a wide range of weights.

Conclusion

Through innovative design and rigorous testing, our team successfully delivered a functional sorting system that autonomously detects and sorts objects by size and weight. Working as an interdisciplinary team, we used our diverse skill sets to solve the tasks at hand. Furthermore, our ability to communicate effectively was tested during the presentation phase, where we demonstrated our final product. This showcased our project management and engineering skills and also highlighted our commitment.

References (MicrocontrollerLab, n.d.)

am.projects.forfun. (n.d.). *Simple Product Sorting System With Raspberry Pi and Arduino.* Retrieved from AUTODESK Instructables: https://www.instructables.com/Simple-Product-Sorting-System-With-Raspberry-Pi-an/

e_Darwish. (n.d.). *Color Sorting System: Arduino Based System With Two Belts.* Retrieved from AUTODESK Instructables: https://www.instructables.com/Color-Sorting-System-Arduino-Based-System-With-Two/

Hub, E. (n.d.). *Interfacing 16X2 LCD With PIC Microcontroller*. Retrieved from https://www.instructables.com/Interfacing-16X2-LCD-with-PIC-Microcontroller/

kenm5. (n.d.). *Raspberry PI L298N Dual H Bridge DC Motor*. Retrieved from https://www.instructables.com/Raspberry-PI-L298N-Dual-H-Bridge-DC-Motor/

MicrocontrollerLab. (n.d.). *PIR Senor Interfacing With Pic Microocntroller*. Retrieved from https://www.instructables.com/PIR-Senor-Interfacing-With-Pic-Microocntroller/

sd_projectmaker. (n.d.). *Conveyor Belt Controller.* Retrieved from SCRIBD: https://www.scribd.com/document/80760803/Conveyor-Belt-Controller

theSTEMpedia. (n.d.). *Color Sorting Machine Using Evive- Arduino Based Embedded Platform.* Retrieved from AUTODESK Instructables: https://www.instructables.com/Color-Sorting-Machine-Using-Evive-Arduino-Based-Em/