



Palestine Polytechnic University
College of Information Technology and Computer
Engineering

OMNI-DIRECTIONAL SORTING CONVEYOR

Team members

Yaman Yousef Abu-Radeh
Noor Yaseer Shahateet
Sally Sami Tarwa

Supervisors

Dr. Radwan Tahbob
Eng. Wael Takrouri

Hebron-Palestine

30-May- 2023

Acknowledgment

This project was not carried out without the help and support that all of you are giving to us, we thank you very much for that. The first to start thanking him is the God - the Almighty -, where all thanks at the first and the last are only for the God - the Almighty -, with no partner of that.

Then, we thank all those who helped us from the academic staff in the Department of Computer System Engineering, especially our supervisors: Dr. Radwan Tahbob and Eng. Wael Takrouri, who were giving us all the advice that we needed while studying the subject of this project.

Finally, we thank all those who participated in the success of this project and did not mention their names here, but our hearts keep thanks and gratitude to them, especially our families and friends.

Abstract

In light of today's technological development, it has become necessary to use machines to reduce the number of workers in factories and delivery companies. For this reason, we decided to work on a sorting conveyor that helps delivery companies sort packages. From our point of view, this project will help the delivery companies sort the packages easier and in less time without the need for many workers to do this task. The system can be used in factories and other industrial distribution centers where the packages must be sorted to take them to their destination. The packages are placed on a conveyor system where the QR on the top of the package is scanned and then it is weighed and classified as (heavy, medium, or light). Depending on the QR and the weight of the packages, they are routed using omnidirectional wheels to their destination.

Keywords: Delivery Companies, Sorting Conveyor, Conveyor System, QR, Omnidirectional Wheels, Path Planning .

الملخص

في ظل التطور التكنولوجي اليوم، أصبح من الضروري استخدام الآلات لتقليل عدد العاملين في المصانع وشركات التوصيل. لهذا السبب، قررنا العمل على ناقل فرز يساعد شركات التوصيل على فرز الطرود. من وجهة نظرنا، يساعد هذا المشروع شركات التوصيل على فرز الطرود بشكل أسهل وفي وقت أقل دون الحاجة إلى العديد من العمال للقيام بهذه المهمة. من أهم أهداف المشروع تسهيل مهمة فرز الطرود بأقل التكاليف وبأقل عدد من العمال عن طريق القيام بذلك بشكل آلي. يمكن استخدام النظام في المصانع ومراكز التوزيع الصناعية الأخرى حيث يجب فرز الطرود لنقلها إلى وجهتها. توضع الطرود على نظام ناقل حيث يتم مسح رمز الاستجابة السريعة ومن ثم قياس وزن الطرود وتصنيفها على أنها (ثقيلة، متوسطة، خفيفة). اعتماداً على رمز الاستجابة و وزن الطرود، يتم توجيهها باستخدام العجلات إلى وجهتها.

الكلمات المفتاحية: شركات التوصيل، ناقل الفرز، نظام ناقل، رمز الاستجابة ، عجلات متعددة الاتجاه ، رسم/تخطيط المسار.

Table of Contents

Abstract	i
List of Figures:	v
List of Tables:	vi
List of Acronyms:	vii
Chapter 1: Introduction	1
1.1 Preface	1
1.2 Project Aims and Objectives	1
1.3 Problem Statement	1
1.4 Project Requirements	2
1.5 System Description	2
1.6 Project Limitations/constraints	3
1.7 Project Schedule	4
1.8 Report Outline	4
Chapter 2: Background	5
2.1 Overview	5
2.2 Theoretical Background	5
2.2.1 Omni-Directional Wheels	5
2.2.2 Weight Sensor	6
2.2.3 QR Code	6
2.3 Literature Review	7
2.3.1 Design and Implementation of Automatic Sorting Machine	7
2.3.2 Potato Sorting Machine Based on LabVIEW with Image Processing strategy.....	8
2.3.3 Development of an Automatic Sorting Machine based on product color and weight characteristics with Conveyor Belt and sensors.....	9
2.3.4 Automatic Sorting Machine Using Conveyor	9
2.4 System's Components	11
2.4.1 Hardware Component	11
2.4.1.1 Processing Node	11
2.4.1.2 Camera Module1	16
2.4.1.3 Motors	17
2.4.1.4 Wheels	19
2.4.1.5 Weight Device	21
2.4.1.6 Other Components	22
2.4.2 Software Component	23
2.5 Summary	24
Chapter 3: Design	22
3.1 Preface	25
3.2 General Block Diagram	25
3.3 Design Option	26
3.3.1 First Design Using Omnidirectional wheels	26
3.3.2 Second Design Using Conveyor Belt	26
3.4 Detailed Block Diagram	27
3.5 System Components	28

3.5.1 Hardware Component	28
3.5.2 Software Component	29
3.5.3 Design of Omni-Directional Wheels	29
3.6 Algorithms and methodologies	30
3.6.1 Flowchart	30
3.7 Schematic diagrams	32
3.7.1 Weight Sensor with HX711	32
3.7.2 ESP 325 Hiletgo Wi-Fi Module with Driver Bridge L298.....	33
3.8 Summary	34
Conclusion	35
Chapter 4: Hardware and Software Implementation	36
4.1 Overview	36
4.2 Hardware Implementation	36
4.2.1 Wheel Implementation	36
4.2.2 Cell Implementation	38
4.2.3 DC Motor Connection	38
4.2.4 Load Cell Connection	39
4.2.5 Path Planning Sketch	40
4.2.6 Download Operating System in Raspberry Pi	42
4.3 Library installation	43
4.4 Software Implementation	43
4.4.1 OpenCV Setup	43
4.4.2 ZBar Setup	43
4.4.3 Hx711 Setup	43
4.4.4 QR Code Generation	44
4.5 Implementation Issues and Challenges	44
Chapter 5: Result and Validation	46
5.1 Overview	46
5.2 Hardware Testing	46
5.2.1 Testing Raspberry Pi	46
5.2.2 Testing Raspberry Pi Camera	46
5.2.3 Testing Load Cell	47
5.2.4 Testing DC Motor and Wheel	47
5.3 Software Testing	48
5.3.1 Testing QR Reader	48
5.4 Integration Testing	49
5.5 Testing Scenario and Result	49
Chapter 6: Conclusion and Future Work	50
6.1 Overview	50
6.2 Conclusion	50
6.3 Future Work	50
References	51

List of Figures

Figure 1.1: Sketch the System	3
Figure 2.1: Omni-Directional Wheels	6
Figure 2.2: Weight Sensor	6
Figure 2.3: QR Code	7
Figure 2.4: Automatic Sorting Machine	8
Figure 2.5: Potato Sorting Machine Based on LabVIEW	8
Figure 2.6: Automatic Sorting Machine Based on Product Color and Weight Characteristics	9
Figure 2.7: Automatic Sorting Machine Using Conveyor	10
Figure 2.8: Raspberry Pi 4	12
Figure 2.9: ASUS Tinker Board 2S	13
Figure 2.10: ESP 32	14
Figure 2.11: Arduino Mega	14
Figure 2.12: Raspberry Pi Camera Module V2.1	16
Figure 2.13: Webcams	17
Figure 2.14: DC Motor	18
Figure 2.15: Stepper Motor	18
Figure 2.16: Omni-Directional Wheels	19
Figure 2.17: Omni-Directional Wheel Direction	20
Figure 2.18: Mecanum Wheels	20
Figure 2.19: Mecanum Wheel Direction	21
Figure 2.20: Digital Scale Blog	22
Figure 2.21: HX711	23
Figure 2.12: L298N	23
Figure 3.1: General Block Diagram	25

Figure 3.2: Second Design Using Conveyor Belt	27
Figure 3.3: Detailed Block Diagram	28
Figure 3.4: Design of Omni-Directional Wheels	29
Figure 3.5: Flowchart of System Activity	31
Figure 3.6: Schematic Diagram for HX711 and Weight Sensor Connected with Raspberry Pi	32
Figure 3.7: Schematic Diagram for ESP 325 Hiletgo Wi-Fi Module Connected with Driver Bridge L298	34
Figure 4.1: Omnidirectional Wheel After Assembling	37
Figure 4.2: Cells Board	38
Figure 4.3: DC Motors Connected with Raspberry Pi	39
Figure 4.4: Load Cell Connection with Raspberry Pi	39
Figure 4.5: The Path for The Unsorted Area	40
Figure 4.6: The Path for The Hebron Light Area	41
Figure 4.7: The Path for The Hebron Heavy Area	41
Figure 4.8: The Path for The Ramallah Light Area	42
Figure 4.9: The Path for The Ramallah Heavy Area	42
Figure 4.10: QR code for Hebron	44
Figure 4.11: QR code for Ramallah	44
Figure 5.1: Camera Connected with Raspberry Pi	46
Figure 5.2: Enabling The Camera Interface from Raspberry Pi Configuration	47
Figure 5.3: Scan The QR Code for Hebron	48
Figure 5.4: Scan The QR Code for Ramallah	48

List of Tables

Table 1.1: Project Schedule	4
Table 2.1: The Differences Between our Project and the Previous Works	11
Table 2.2: Difference Between Raspberry Pi 4 and ASUS Tinker Board	13
Table 2.3: Difference Between ESP 32 and Arduino Mega	15
Table 2.4: Difference Between Pi Cameras and Webcams	17
Table 2.5: Difference Between DC Motor and Stepper Motor	19
Table 2.6: Difference Between Weight Sensor and Digital Scale.....	22
Table 4.1: Pieces for One Wheel	37
Table 4.2: Total Time and Filament Need to Print One Wheel	45
Table 5.1: Testing Scenario and Result	49

List of Acronyms

CSI	Camera Serial Interface
DC	Direct Current
DSI	Display Serial Interface
HDMI	High-Definition Multimedia Interface
GPIO	General Purpose Input/output
MIPI	Mobile Industry Processor Interfaced
PLC	Programmable Logic Controller
QR	Quick Response
SBC	Single Board Computer
SD	Secure Digital
SOC	System On Chip
UAV	Unmanned Aerial Vehicle
USB	Universal Serial Bus
MIPI	Mobile Industry Processor Interface
CSI	Camera Serial Interface
ESP	Extra Sensory Perception

Chapter 1

Introduction

1.1 Preface

The conveyor system is a fast and efficient mechanical handling and is used in many industries to transport goods, products, raw commodities, and other materials from one place to another. They appear in many applications including industrial processing plants, warehousing, freight handling, and other transportation operations. In our project, a conveyor system will be used to move packages after they are weighed by a device that measures weight and also to classify them to their destination cities and drop them to their desired sorting areas.

1.2 Project Aims and Objectives

- Design and implementation of automated sorting conveyor. The packages are sorted automatically without the need for human intervention. Automated sorting conveyor reduces time and effort from the sorting process.
- Measure the weight of packages using a weight device and classify these packages into different ports based on weight ranges (light - heavy).

1.3 Problem Statement

Many delivery companies and factories face difficulties in sorting packages, as it is possible to reach thousands of packages on the same day, which need to be sorted according to the destination as well as according to weight in order to reach correctly and efficiently to the desired destination, this can be done by hand, but it will involve large numbers of employees in addition scanning, weighing and separating multi-delivery packages in the future, and these tasks are repetitive and fast and take a long time, which costs time and effort and costs high labor costs and it is possible

that an error may occur in the sorting process, packages sorting technology reduces the burden on employees and to solve this problem we use mechanism that can scan packages and calculate their weight. In summery this project solves the problem of weighing and classifying the packages to their desired destination.

1.4 Project Requirements

1.4.1 Functional Requirements:

The System must be able to:

- Read the QR on the packages using the camera.
- Weigh the packages using a weighing system.
- Move packages using omnidirectional wheels toward desired sorting areas, as these wheels have the ability to rotate 360 degrees.

1.4.2 Non Functional Requirements:

- Availability: The system will remain active all the time as long as there is an energy source .
- Scalability: The system can be extended, it is possible to increase the number of cities and the number of sorting packages.
- Efficiency: Instead of sorting packages manually by worker, the system is more efficient, it sorts the packages automatically.

1.5 System Description

The system consists of a board with multi-moving omnidirectional wheels each wheel is attached to a DC motor; it has a camera to give feedback to the system and also it has a weight measuring device to weigh the packages to be sorted. The board consists of a group of cells, each cell containing three motors and for each motor has an omnidirectional wheel attached to it. The camera is located at the top of the board and it reads the QR code for each package affixed to the package from which the city of that package is determined by a library that is downloaded to the Raspberry Pi, which reads

the QR code. As for the device, it is next to the camera on the board, the package moves to a sensor that measures the weight and the system classifies the package to one of the following (Heavy, Light), and depending on its QR code and weight, the package is moved through its assigned cells by working some motors to reach the city and its sorting area, where each city has its own two sorting areas based on weight, or if there is any error, whether in reading the code or measuring the weight of the package, the package will be directed to an area as shown in the Figure 1.1.

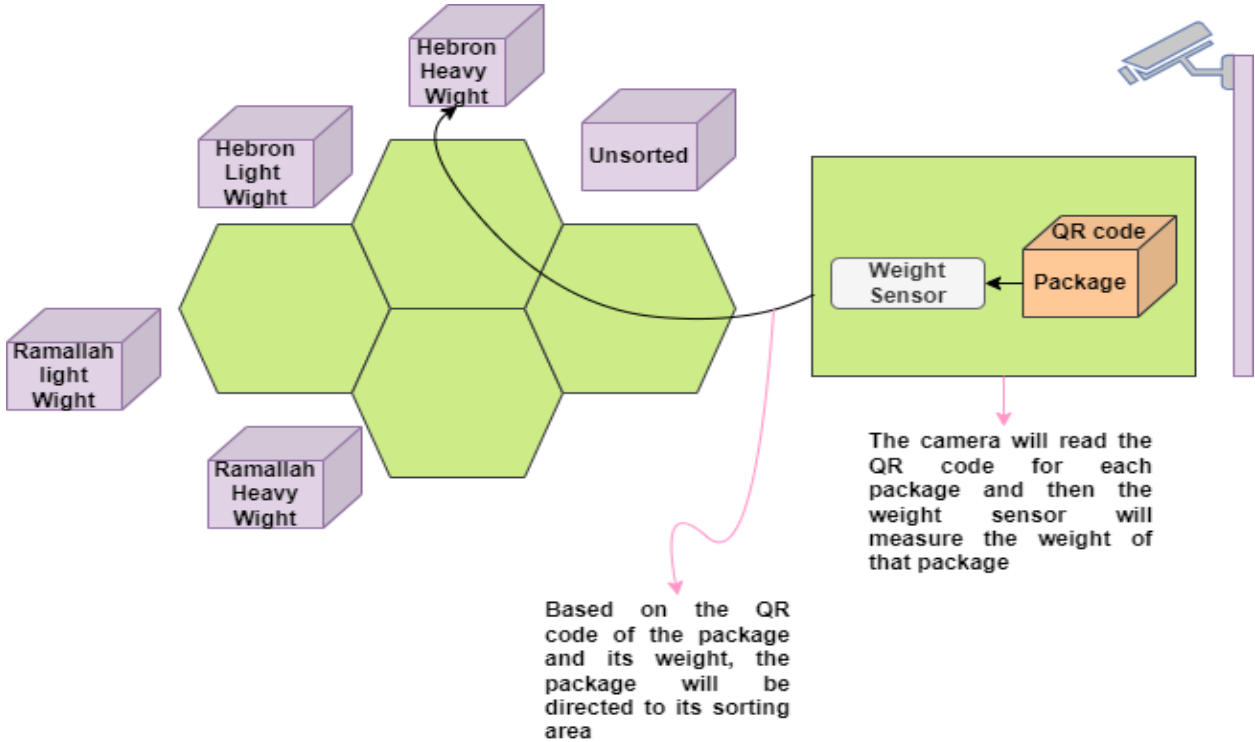


Figure 1.1: Sketch of The System

1.6 Project Limitations/Constraints

- The package must be in the shape of a box, not a bag.
- Only one package can be sorted at a time .
- The dimensions of the package must be 25x25x25 cm or less.

- The weight of packages must be within what the motor can move. it must not exceed 1 kg, If the weight is less than 500 g, it is classified as light, and if it is greater than 500 g to 1,000 g, it is classified as heavy.
- Each package must have an identification number or QR code that identifies the target area.

1.7 Project Schedule

Task	Duration (weeks)															
	First Semester								Second Semester							
	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32
<i>Project idea</i>	■	■														
<i>Project planning</i>			■	■												
<i>Analyzing & design</i>					■	■	■									
<i>Project development</i>								■	■	■	■	■	■	■		
<i>Project test</i>															■	■

Table 1.1: Project Schedule

1.8 Report Outline

This report is organized as Chapter one gives an introduction to the system. Chapter two discusses literature reviews, it also goes briefly over the theoretical background of the project, hardware, and software components. Chapter three contain design options, block diagrams, flowchart and schematic diagrams. Chapter four contain the software and hardware implementation. Chapter five contain the result and validation. Finally, Chapter six contain conclusion and Future work.

Chapter 2

Background

2.1 Overview

This chapter describes the theoretical background and the literature review of the project and presents hardware and software components that will be used in system design and implementation.

2.2 Theoretical Background

A conveyor system is a common piece of mechanical handling equipment that moves materials from one location to another. Conveyors are especially useful in applications involving the transport of heavy or bulky materials. Conveyor systems allow quick and efficient transport for a wide variety of materials, which make them very popular in the material handling and packaging industries. They also have popular consumer applications, as they are often found in supermarkets and airports, constituting the final leg of item/ bag delivery to customers. Many kinds of conveying systems are available and are used according to the various needs of different industries. There are chain conveyors (floor and overhead) as well. In our project, a conveyor system will be used to move packages and classify them to their destination cities and drop them to their desired sorting areas using omnidirectional wheels.

2.2.1 Omni-Directional Wheels

Omni-Directional wheels roll forward like normal wheels but slide sideways with almost no friction and no skidding during turns. The angled rollers on these wheels allow vehicles (or packages on them) to move forward, backward, sideways and turn in place

when each wheel is used with an independent drive. A left and a right wheel must be paired together to use all of the movement combinations, it can perform 360-degree maneuverability and rotation functions, including side movement and diagonal movement. They are often used in intelligent robot research for small autonomous robots. In projects such as VEX robotics, RoboCup robotics, and FIRST robotics, many robots use these wheels to have the ability to move in all directions. In our project, a group of these wheels will move together to deliver the box to the distribution regions and their sorting areas throughout the country. The Omni-Directional wheel is shown in Figure 2.1. [1]



Figure 2.1: Omni-Directional Wheels. [2]

2.2.2 Weight Sensor

A weight sensor shown in Figure 2.2 is a particular kind of transducer, more especially a weight transducer, by definition. And it is a device used to measure force and load, to measure weight, sensors used internal strain gauges Metal foil strain gauge is a sensor whose electrical resistance varies with applied force. In other words, it converts (or transduces) force, pressure, tension, compression, torque, weight, etc... into a change in electrical resistance, which can then be measured. [3]

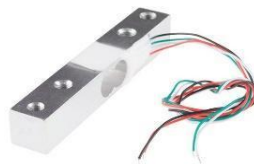


Figure 2.2: Weight Sensor. [4]

2.2.3 QR Code

QR code was invented in Japan in 1994 by a development team led by Masahiro Hara for the company Denso Wave a subsidiary of the Japanese automotive company Toyota, Hara is widely credited as the creator of the QR code. Hara's team's task was to create a barcode that could easily track automobiles and automobile parts during manufacturing. They invented the QR code, which was designed to be read quickly and easily by scanners, and it soon became popular outside of the automotive industry as a way to track inventory and product information. They did not expect it to be used outside of the automotive industry., In 2000, the QR codes became internationally standardized by ISO/IEC standards. QR codes shown in Figure 2.3 are the natural evolution of the barcode. It is characterized by the speed at which they can be scanned, and it is a scan-able barcode encoded with data. Encoded means converted into a particular form. In the case of QR codes, numeric and alphanumeric characters, bytes, and kanji convert (one of the two systems used to write the Japanese language) into a unique two-dimensional arrangement of squares. When the box goes to the bottom of the camera and takes a scan over those squares, it translates their arrangement back into that data's original form, QR Code consists of 7 parts, and each of these parts creates a sort of pixel pattern that looks similar to a crossword puzzle. Each element has a specific purpose that conveys certain information through the code to determine in the end this address is specific to a specific area throughout the country. [5][6]

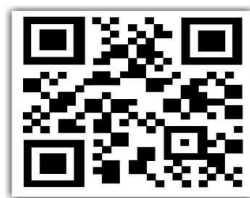


Figure 2.3: QR Code. [7]

2.3 Literature Review

The following are some projects and some studies published in international papers and journals that are related to our project.

2.3.1 Design and Implementation of Automatic Sorting Machine

In this project the authors (Islam Abu Khalaf, Murad Abu Arfeh) designed and implemented a machine shown in Figure 2.4 that sorts and classifies three different types of materials which are iron, aluminum, and plastic. Then guide materials to different carts to prepare for the material recycling process. Materials are gravity-fed from the hopper to the conveyor. Then the entered materials are scanned by sensors (inductive and magnetic) that are connected to a programmable logic controller (PLC) which gives the air pistons that are located at the edges of the conveyor to move each type of material to a specific cart. [8]



Figure 2.4: Automatic Sorting Machine. [8]

2.3.2 Potato Sorting Machine Based on LabVIEW with Image Processing Strategy

This project is done by Anas Al-Harine and Diaa Musalm. It is aimed at providing the local market with fully automated sorting machines capable of sorting different sizes of vegetables and fruits. The proposed technology compares the object size with the already installed image by using an image camera integrated LabVIEW platform. As a result of the applied technology and control strategy, the system is characterized by production rate and acceptable accuracy which in turn avoids the probability of human errors and inaccuracy. The Potato sorting Machin is shown in Figure 2.5. [9]

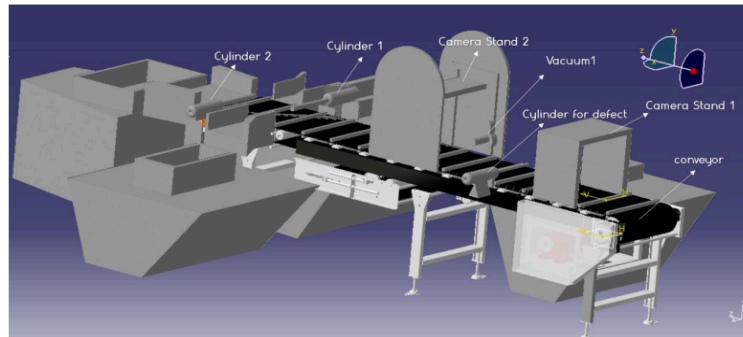


Figure 2.5: Potato Sorting Machine Based on LabVIEW. [9]

2.3.3 Development of an Automatic Sorting Machine Based on Product Color and Weight Characteristics with Conveyor Belt and Sensors.

This study is done by [Y.M. Anekar, V.S. Koli, S.D. Degloorkar, S.B. Bharamgonda, Y. R. Mahulkar]. In this study, the products are sorted based on color and weight. A single conveyor belt shown in Figure 2.6 is used and it is controlled by DC motors. This belt contains a color sensor and a weight sensor at the beginning of it in order to separate products on the basis of color and weight.

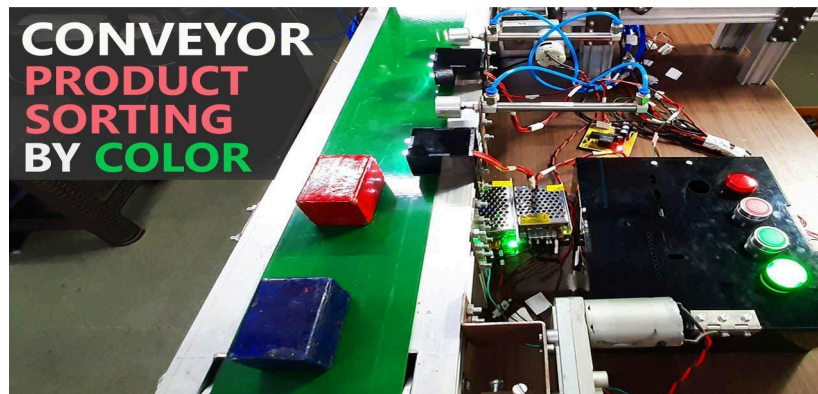


Figure 2.6: Automatic Sorting Machine Based on Product Color and Weight Characteristics. [10]

2.3.4 Automatic Sorting Machine Using Conveyor

This project is done by Arena company. It uses proximity sensors to distinguish three types of materials (wood, plastic, and metal). The objects are also sorted concerning their height on three different levels and sent to their respective compartments using the pushing mechanism (conveyor belt, pneumatic cylinder, and PLC). As shown in Figure 2.7. [11]

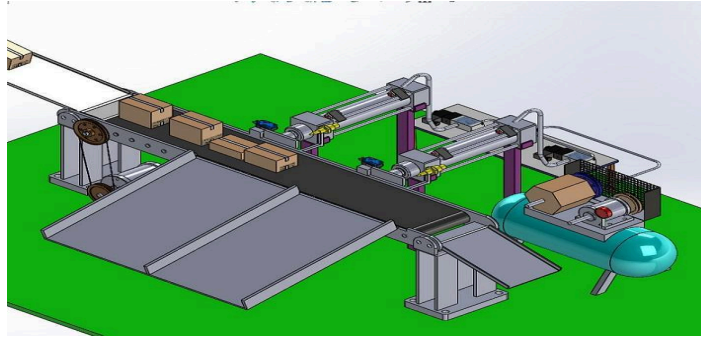


Figure 2.7: Automatic Sorting Machine Using Conveyor. [11]

The differences between our project and the previous works are summarized in Table 2.1. It shows the following aspects:

#	Sorting basis	Mechanisms used in sorting	Weighing Sensor	Motors type
Omni-Directional Motor	Sorting packages based on their location and its weight	Image processing (using QR scanning)	Contains weighing sensor	Omni-directional motor
Design and Implementation of Automatic Sorting Machine	Sorting minerals based on their type	Using inductive and electromagnetic sensors	Doesn't contain weighing sensor	DC motor
Potato Sorting Machine Based on LabVIEW With	Sorting potato based on their size	Image processing based on LabVIEW	Doesn't contain weighing sensor	DC motor

Image processing Strategy				
Development of an Automatic Sorting Machine Based on Product Color and weight Characteristics with Conveyor Belt and Sensors	Sorting products based on color and weight	Using color sensor	Contain weighing sensor	DC motor
Automatic Sorting Machine Using Conveyor	Sorting products based on their materials	Using pushing mechanisms (conveyor belt, Pneumatic cylinder, and PLC)	Contain weighing sensor	DC motor

Table 2.1: The Differences Between Our Project and The Previous Works.

2.4 System's Components

This section contains a short description of the hardware and software components used in our system. It also contains a brief justification for each certain choice.

2.4.1 Hardware Component

The system will use a processing node, camera module, motors, Wheels, and weight device. The following section will list the previous components, options, and specifications.

2.4.1.1 Processing Node

2.4.1.1.1 Raspberry Pi 4

The Raspberry Pi 4 is a low-cost, credit-card-sized computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing and learn how to program in languages like Scratch and Python. It's capable of doing everything expected a desktop computer to do, from browsing the internet and playing high-definition video, to making spreadsheets, word processing, and playing games.

The Raspberry Pi is a series of small single-board computers developed in the United Kingdom by the Raspberry Pi Foundation to promote the teaching of basic computer science in schools and in developing countries. The original model became far more popular than anticipated, selling outside its target market for uses such as robotics. It does not include peripherals (such as keyboards and mice) or cases. However, some accessories have been included in several official and unofficial bundles. [13]

The Raspberry Pi 4 Model B, shown in Figure 2.8 was launched in June 2019. It uses a 1.5GHz 64-bit quad-core Arm Cortex-A72 CPU, has three RAM options (2GB, 4GB, 8GB), gigabit Ethernet, integrated 802.11ac/n wireless LAN, and Bluetooth 5.0. The new Raspberry Pi 4 has upgraded USB capacity: along with two USB 2 ports and has two USB 3 ports, which can transfer data up to ten times faster. [13]



Figure 2.8: Raspberry Pi 4 [12]

ASUS Tinker Board

The ASUS Tinker Board, shown in Figure 2.9 is a Single Board Computer (SBC) in an ultra-small form factor that offers class-leading performance while leveraging outstanding mechanical compatibility. The Tinker Board offers makers, IoT enthusiasts, hobbyists, PC DIY enthusiasts, and others a reliable and extremely capable platform for building and tinkering their ideas into reality. [18]



Figure 2.9: ASUS Tinker Board 2S [19]

Table 2.2 presents the difference between ASUS Tinker Board and Raspberry Pi, and because of this we choose Raspberry Pi.

Feature	Raspberry Pi	ASUS Tinker Board
Processor	Broadcom BCM2711, Quad-core Cortex-A72(ARMV8) 64-bit SoC @ 1.5GHz	Rock chip RK3288 1.8Ghz Quad-Core ARM Cortex-A17-32 bit
RAM	1GB,2GB or 4GB LPDDR4-3200 SDRAM (depending on model)	2GB LPDDR3
GPU	Broadcom VideoCore VT @ 500MHz	Mali T764 @ 600 MHz
Display	2 x micro-HDMI ports supporting 4K@60Hz displays via HDMI 2.0,MIPI DSI display port	HDMI with 11.264 and H.265 4K decode capability
Ports	Dried 4 x USB 2.0 40 x GPIO 1 x HDMI 1 x Camera Serial Interface (CSI) 1 x Display Serial Interface (DSI) 3.5mm audio jack	2 x USB 3.0 2 x USB 2.0 40 x GPIO 1 x HDMI 2-lane MIPI Camera Serial Interface (CSI) 2-lane MIPI Display Serial Interface (DSI) 3.5mm audio-video jack
Storage	Micro-SD+USB boot mode	Micro-SD
Cost	\$141. [25]	\$159. [26]

Table 2.2: Difference Between Raspberry Pi 4 and ASUS Tinker Board

2.4.1.1.2 ESP 32

The ESP32 shown in Figure 2.10 is a widely used, highly integrated microcontroller and Wi-Fi/Bluetooth system-on-chip (SoC) designed for Internet of Things (IoT) applications. It is developed by Espressif Systems and offers a powerful combination of features, making it popular among developers and hobbyists.

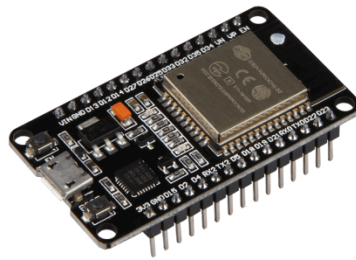


Figure 2.10 :ESP 325 [20]

Arduino Mega

The Arduino Mega shown in Figure 2.11 is an open-source microcontroller board based on the ATmega2560 microcontroller. It is an enhanced version of the Arduino Uno board and is designed to provide more input/output (I/O) pins and memory for complex projects that require a higher level of processing power.



Figure 2.11: Arduino Mega [21]

Table 2.2 presents the difference between ESP 32 and Arduino Mega, and because of this we choose ESP 32.

Feature	ESP32	Arduino Mega
Microcontroller	Dual-core Xtensa LX6, up to 240 MHz	ATmega2560, 16 MHz
Wireless Connectivity	Built-in Wi-Fi and Bluetooth capabilities	No built-in wireless connectivity
GPIO Pins	38 GPIO pins	54 digital I/O pins and 16 analog input pins
Memory	Up to 520 KB SRAM and up to 4 MB Flash memory	8 KB SRAM and 256 KB Flash memory
Processing Power	Higher clock speed and dual-core architecture	Lower clock speed and single-core architecture
Price	Generally more affordable	Generally more expensive

Table 2.3: Difference Between ESP 32 and Arduino Mega.

2.4.1.2 Camera Module

Pi Camera Module

The Pi camera module is a portable lightweight camera that supports Raspberry Pi. It communicates with Pi using the MIPI (Mobile Industry Processor Interface) camera serial interface protocol. It is normally used in image processing, machine learning or in surveillance projects. It is commonly used in surveillance drones since the payload of the camera is very less. Apart from these modules, Pi shown in Figure 2.10 can also use normal USB (Universal Serial Bus) webcams that are used along with a computer. MIPI modules are ideal for multi-camera applications including mobile and distributed applications like autonomous driving, UAV (Unmanned Aerial Vehicle), Smart City, medical technology, and laboratory automation. [13]

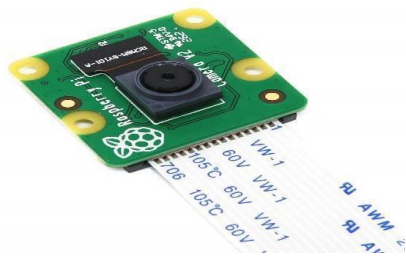


Figure 2.12: Raspberry Pi Camera Module V2.1 [14]

Webcam

A webcam is a small digital video camera directly or indirectly connected to a computer or a computer network. Webcams come with software that needs to be installed on the computer to help users record video on or stream it from the Web. Webcams, as shown in Figure 2.11 are capable of taking pictures as well as high-definition videos, although the video quality can be lower compared to other camera models. [27]



Figure 2.13: Webcams [28]

Table 2.3 presents the difference between Pi Camera and Webcams, and because of this we choose Pi Camera.

Feature	Pi Camera	Webcams
Mount Type	A stylish, small, and adjustable mount	fixed and flexible
Resolution	5 megapixel	2073600 pixels or 2.0736 megapixels
Angle of View	62.2 degrees	60-degrees captures one person sitting in front of a compute
Application	UAV(Unmanned Aerial Vehicle), Smart City, medical technology, and laboratory automation	video telephony, live streaming and social media, and Security

Table 2.4: Difference Between Pi Camera and Webcams

2.4.1.3 Motors

DC Motor

A DC motor is any of a class of rotary electrical machines that converts direct current electrical energy into mechanical energy. The most common types rely on the forces produced by magnetic fields. Nearly all types of DC motors have some internal mechanism, either electromechanical or electronic, to periodically change the direction of current in a part of the motor. A DC motor shown in Figure 2.12 consists of a stator, an armature, a rotor, and a commutator with brushes. The opposite polarity between the two magnetic fields inside the motor causes it to turn. DC motors are the simplest type of motor and are used in household appliances, such as electric razors, and in electric windows in cars. [13]



Figure 2.14: DC Motor [15]

Stepper Motor

Stepper motors are DC motors that move in discrete steps. They have multiple coils that are organized in groups called "phases". By energizing each phase in sequence, the motor will rotate, one step at a time. Essentially, stepper motors offer excellent speed control, precise positioning, and repeatability of movement. Additionally, stepper motors shown in Figure 2.13 are highly reliable since there are no contact brushes in the motor. This minimizes mechanical failure and maximizes the operation lifespan of the motor. [20]

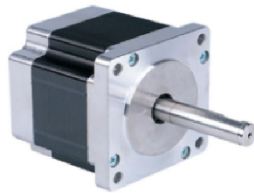


Figure 2.15: Stepper Motor [21]

Table 2.4 presents the difference between DC Motor and Stepper Motor, and because of this we choose DC Motor.

Basis of Difference	Stepper Motor	DC Motor
Efficiency	The efficiency of stepper motors is low.	The DC motors have high efficiency around 85%.
Motion	Stepper motors have incremental motion.	DC motors have continuous motion.
Service Type	The stepper motors are not meant for continuous use because they tend to run hot	DC motors are designed to operate continuously without much problem.

	when powered for a long duration.	
Response	Stepper motors give slow response.	A DC motor equipped with feedback control can give a much faster response than a stepper motor.

Table 2.5 Difference Between DC Motor and Stepper Motor

2.4.1.4 Wheels

Omni-Directional Wheel

Omni-Directional wheels, shown in Figure 2.14 are rolled forward like normal wheels but slide sideways with almost no friction and no skidding during turns. The angled rollers on these wheels allow vehicles (or packages on them) to move forward, backward, sideways and turn in place when each wheel is used with an independent drive. A left and a right wheel must be paired together to use all of the movement combinations; it can perform 360-degree maneuverability and rotation functions, including side movement and diagonal movement. The Omni-directional wheel direction is shown in Figure 2.15. [1]



Figure 2.16: Omni-Directional Wheels [29]

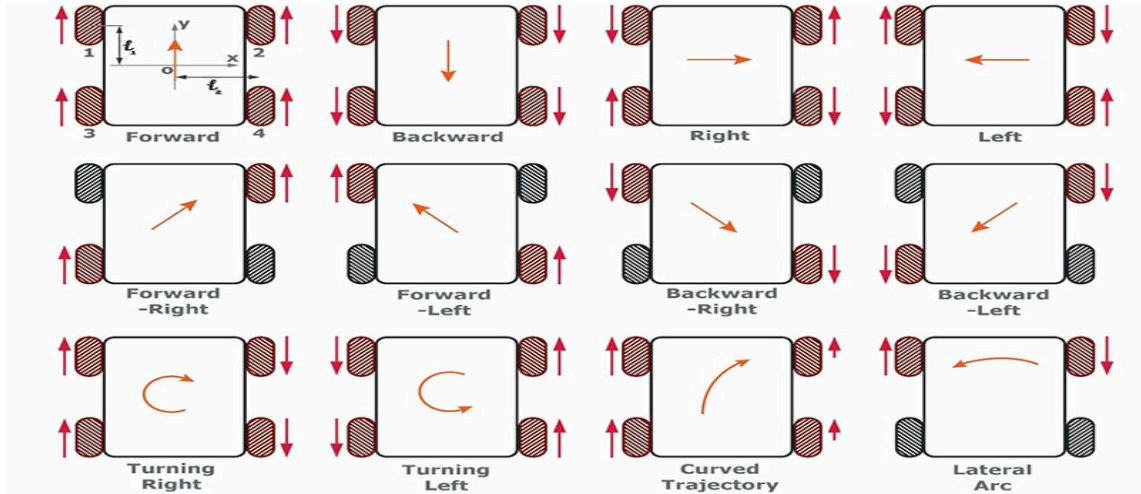


Figure 2.17: Omni-Directional Wheel Direction [22]

Mecanum Wheel

A Mecanum wheel, shown in Figure 2.16 is a standard wheel to which a set of rollers have been attached, each of which rotates about an axis 45 degrees to the plane of the rotation. Mecanum wheel-fitted robots or industrial machines can make a 180-degree turn without taking up much space. Mecanum wheels are shown in Figure 2.17. [23]



Figure 2.18: Mecanum wheels [30]

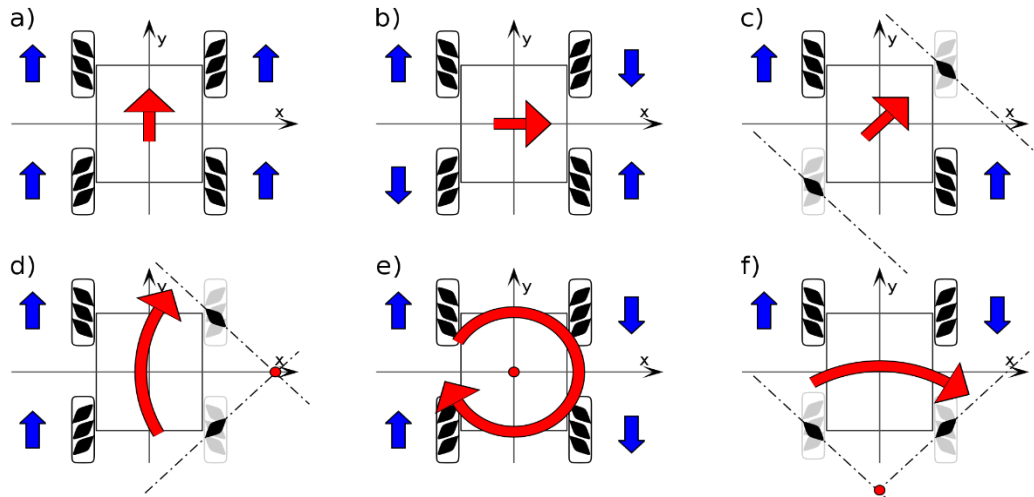


Figure 2.19: Mecanum Wheel Direction [24]

Between the mecanum wheel and Omni-Directional wheel, we chose an Omni-Directional wheel because it rotates 360 degrees in eight directions while the mecanum rotates 180 degrees in six directions.

2.4.1.5 Weight Device

Weight Sensor

A weight sensor is a device used to measure force and load. It converts weight into an electrical signal which can be processed and used within various applications. To measure weight, sensors used internal strain gauges to measure the weight. The weight sensor is shown in Figure 2.2. [3]

Digital Scale Blog (EK-i/EW-i Series)

Digital Scale Blog It is a sanitary and chemically resistant weighing container made of stainless steel (SUS304) It has the ability to adjust response to help deal with the effect of drafts and vibrations It is able to measure multiple weighing units, including g, oz, lb, ozt, ct, mom, dwt, its function is mode percentage, auto power off and many other functions. Digital Scale Blog is shown in Figure 2.18. [31]



Figure 2.20: Digital Scale Blog [31]

Table 2.5 presents the difference between weight sensor and Digital scale blog, and because of this we choose weight sensor.

Feature	Weight Sensor	Digital Scale Blog
Model	Micro load cell	EK-300i
Capacity	50Kg	400g
Size	55.2x12.7x12.7mm	4.3x110mm
Cost	\$5.45 [32]	\$379 [33]

Table 2.6: Difference Between Weight Sensor and Digital Scale Blog.

2.4.1.6 Other Components

HX711 Module

HX711 is a precision 24-bit analog-to-digital converter (ADC) designed for weigh scales and industrial control applications to interface directly with a bridge sensor. And its allow to easily read load cells to measure weight. By connecting the amplifier to microcontroller you will be able to read the changes in the resistance of the load cell. [35]

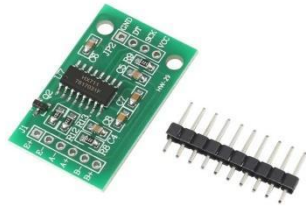


Figure 2.21: HX711 [36]

And The Raspberry Pi does not have an ADC, so we used HX711 to convert analog data to digital data between the raspberry pi and weight sensor

Driver Bridge L298N

The L298N is a popular dual H-bridge motor driver integrated circuit (IC) used for controlling and driving DC motors or stepper motors. It provides a convenient way to control the speed and direction of motors in various robotic and automation applications.

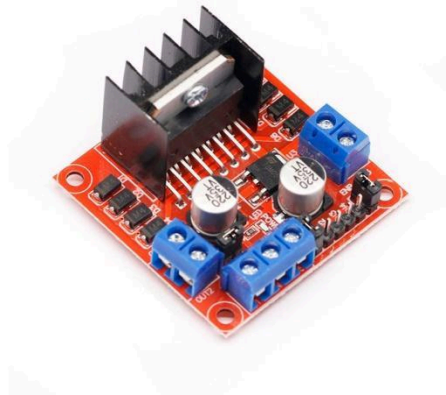


Figure 2.22: L298N [37]

2.4.2 Software Component

Python

Python is a well-known high-level programming language that may be used for a variety of purposes. Python has a dynamic system and memory management that allows a variety of programming paradigms, including object-oriented, functional, and procedural programming. Aside from that, it includes extensive and well-organized standard libraries. Python interpreters are available for many operating systems, allowing Python code to run on a wide variety of systems. [16]

ZBar and OpenCV Library

ZBar is an open source software suite for reading bar codes from various sources, such as video streams, image files and raw intensity sensors. It supports many popular symbologies (types of bar codes) including EAN-13/UPC-A, UPC-E, EAN-8, Code 128, Code 39, Interleaved 2 of 5 and QR Code.

we are going to read a QR Code by Raspberry Pi using OpenCV and ZBar library. ZBar is the best library for detecting and decoding the different types of barcodes and QR codes. OpenCV is used to grab a new frame from a video stream, and process it. Once OpenCV captures a frame, then we can pass it to a dedicated Python barcode decoding library such as a ZBar which decodes the bar code and converts it into respective information. [17]

2.5 Summary

In this chapter, we made a clarification of some theoretical concepts that will be used in our project, including (Omni-directional motor, weight sensor, and QR code) their definition, and method of use. This chapter also lists some of the projects and studies similar to our project and a comparison between them from different aspects such as

(sorting basis, mechanism, and type of motor). Also, we have made a simple clarification of the hardware and software components that will be used in our project.

Chapter 3

Design

3.1 Preface

This chapter presents a general block diagram and some design options, it also covers the overall design of the system and the integration of its components, shows the detailed block diagram, and schematic diagram, as well as details about the algorithms we will use.

3.2 General Block Diagram

The block diagram of Omnidirectional sorting conveyor illustrated which shows the components of the system that contain a controller, weight sensor, scanning module and several motors shows in Figure 3.3.

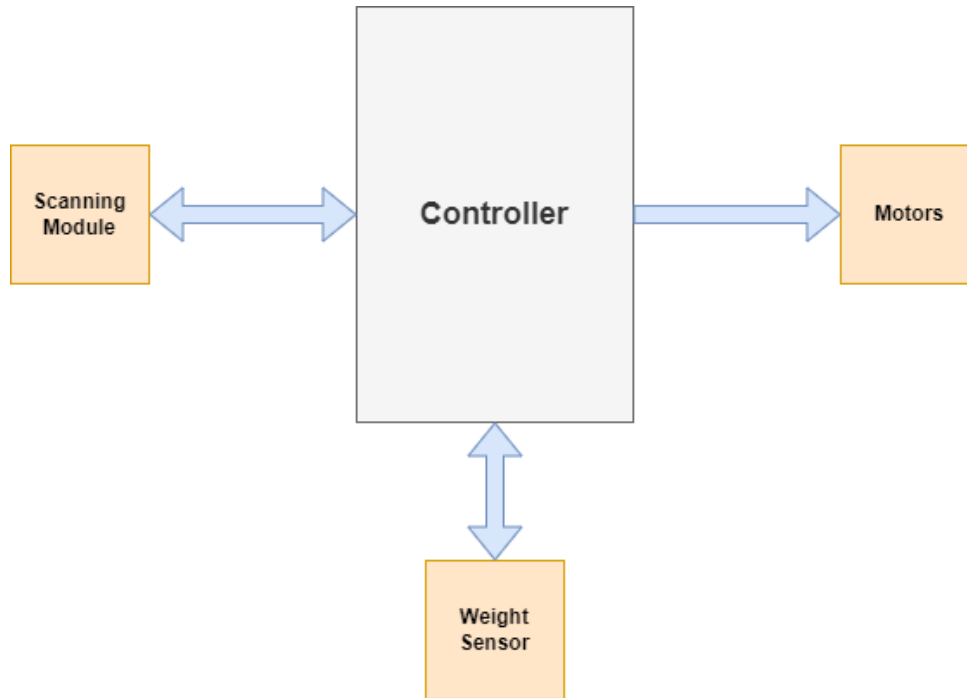


Figure 3.1: General Block Diagram

3.3 Design Options

We had several options regarding how to design the project, and what is the best and most appropriate way to design, and these options are the following:

3.3.1 First Design Using Omnidirectional Wheels

When we use the Omnidirectional wheels, the packages will enter the system, and the camera recognizes the Qr code at the top of the package, then move to the weight sensor, measures its weight, then the package enters the sorting system and The responsible motors work together to move the omnidirectional wheels 360° for deliver the package to its sorting area according to its weight (heavy, light), or if there is any error, whether in reading the code or measuring the weight of the package, the package will be directed to an sorted area as shown in Figure 1.1.

3.3.2 Second Design Using Conveyor Belt

When we use a belt in the system to pass the packages and distribute them, the mechanism will be as follows, after the camera reads the QR code for the packages and passes it to the weight sensor, it will enter the sorting system, there will be a director to determine which belt the package should go to, after that the package enters the desired belt It arrives at the sorting area that it must reach. There will also be another director to specify any sorting area in it according to its weight, and after that is determined, the packages will enter another belt to reach the desired, as shown in Figure 3.2.

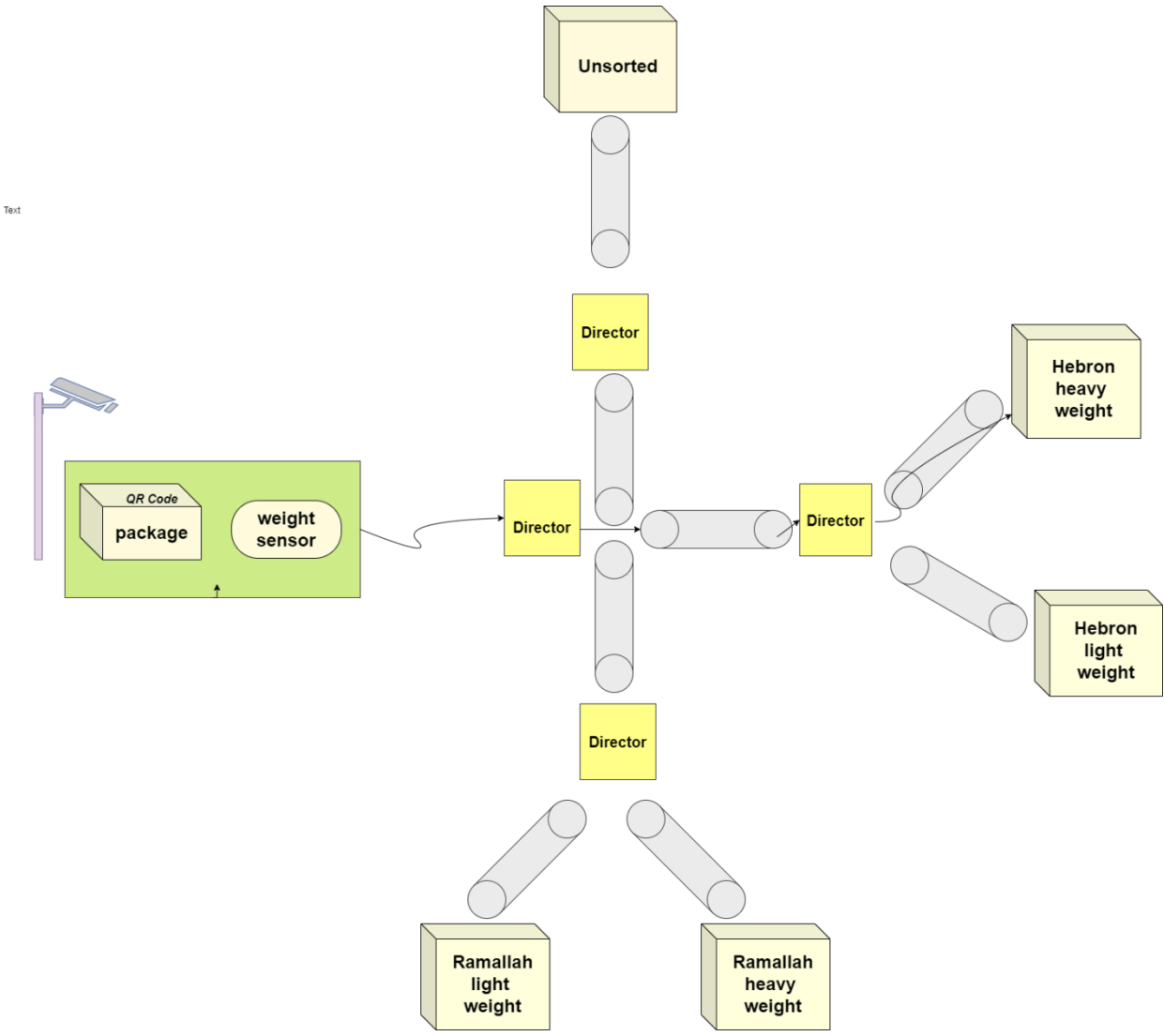


Figure 3.2: Second Design Using Conveyor Belt

3.4 Detailed Block Diagram

The block diagram shows a detailed overview of the main components of the systems and the connection between them. So that there are two units to control the system, the first is the raspberry pi, where it scans the symbol at the top of the package and analyzes it to find out which city it belongs to. The weight sensor measures the weight of the package and sorts it based on its weight to the desired city and directs it through the DC motors. The DC motors are driven by the second ESP32 controller as shown in Figure 3.3.

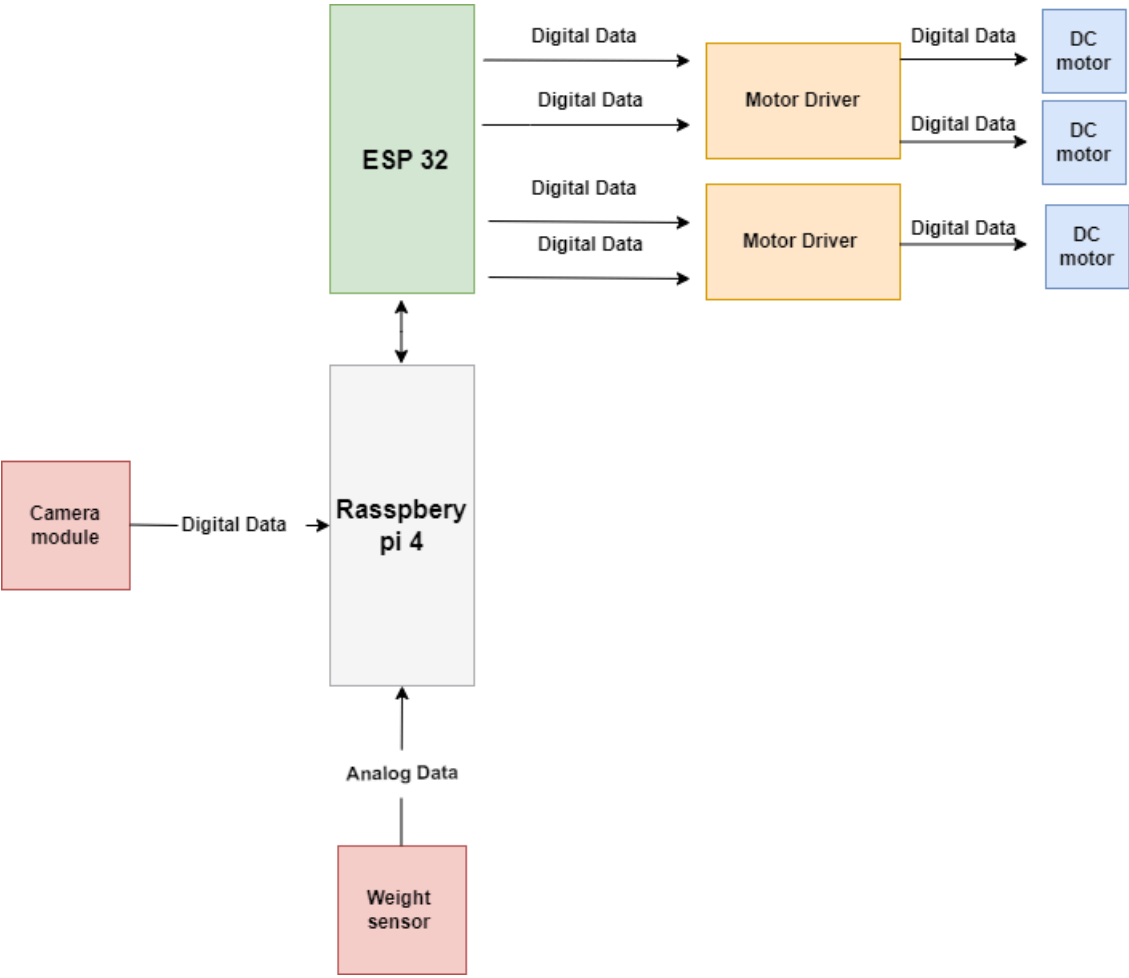


Figure 3.3: Detailed Block Diagram for One Cell

3.5 System Components

3.5.1 Hardware Component

- Processing Node
- Camera Module
- HX711
- Motors
- Wheels
- Weight Device

- L289M

3.5.2 Software Component

- Python
- ZBar and OpenCV library

These components have been discussed in detail in the previous chapter

3.5.3 Design of Omni-Directional Wheels

Due to the unavailability of Omni-Directional Wheels on our local market at the present time, we decided to go to the option of printing the Omni-Directional Wheels with a 3D printer and using what is known as the filament that we need to print the parts of the wheels, and through the designs shown below, we sent them to the printer to print them.

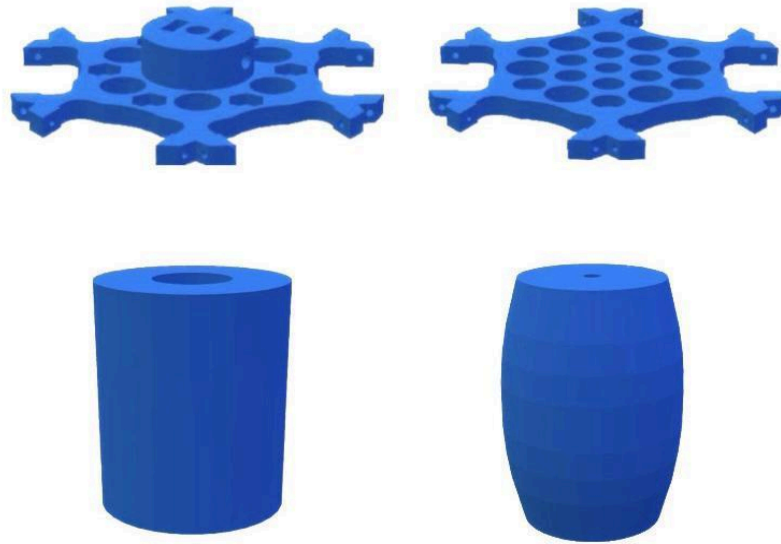


Figure 3.4 : Design of Omni-Directional Wheels.[38]

3.6 Algorithms and Methodologies

3.6.1 Flow Chart

Figure 3.5 represents the flow chart of system activity. When the package enters the system the camera scans the QR code from the package and defines this package following city A or city B or city C. Then the package move to the weight sensor measures its weight and defines the package weight as heavy or medium or light. According to the package city and their weight the omnidirectional wheel starts rotating to move the package the desired sorting area.

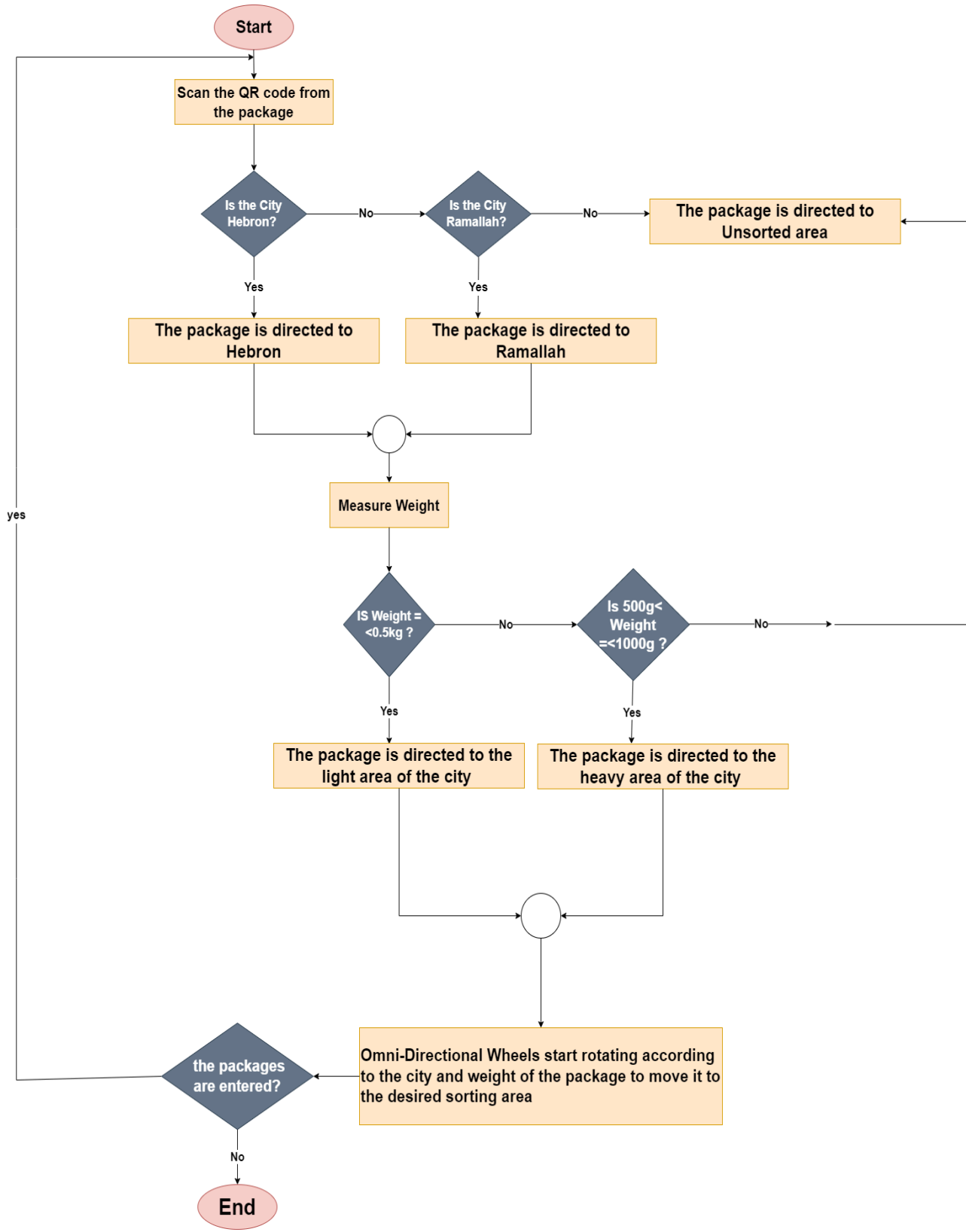


Figure 3.5: Flowchart of System Activity

3.7 Schematic Diagrams

3.7.1 Weight Sensor with HX711

The interfacing circuit between HX711 and weight sensor with Raspberry pi:

First, connect VCC of HX711 to external 5V supply (assuming you are using a weight sensor)

Then connect GND of HX711 to GND of Raspberry pi.

There are 2 pins in HX711 (CK/TX, DT/RX) connected with 2 pins in Raspberry pi (GPIO 6, GPIO 5).

Finally, We have 4 pins in the HX711 (READ, BLACK, GREEN, WHITE) connected with 4 pins in Raspberry pi (E+, E-, A-, A+) as shown in the schematic diagram in Figure 3.6.

Conclusion

What we know now is a visualization of our system architecture. We have demonstrated the problem statement and objectives of our project, described functional and non-functional requirements, and described in detail the system and its limitations. We explained some theoretical concepts that will be used in our project, and listed some projects similar to ours, and compared them. Also, we showed the hardware and software components, semantics diagrams and algorithms that we need to implement our project.

Chapter Four

Hardware and Software Implementation

4.1 Overview


This chapter describes the project software and hardware implementation as well as the different components and tools used to build the System.

4.2 Hardware Implementation

4.2.1 Wheel Implementation

At first, we print omnidirectional wheel pieces using the ready-made 3D design shown in Chapter 3, using a Sindoh (3DW0X1) printer and PLA Filament ANET print material then assemble these pieces using screws, wire, and adhesives to get a complete wheel.

To assemble one wheel, we need the pieces shown in Table 4.1:

#	Name of piece	Quantity	Picture
1	Omni_2stl	1 pcs	








2	Omni_1_5mm_hole.stl	1 pcs	
3	Omni_roller.stl	12 pcs	
4	Omni_spacer.stl	3 pcs	
5	M3 16 mm bolt	3 pcs	
6	M3 12 mm bolt	2 pcs	
7	Axis for rollers	12 pcs	
8	M3 nut	5 pcs	

Table 4.1: Pieces for One Wheel

The wheel after assembling is shown in Figure 4.1:

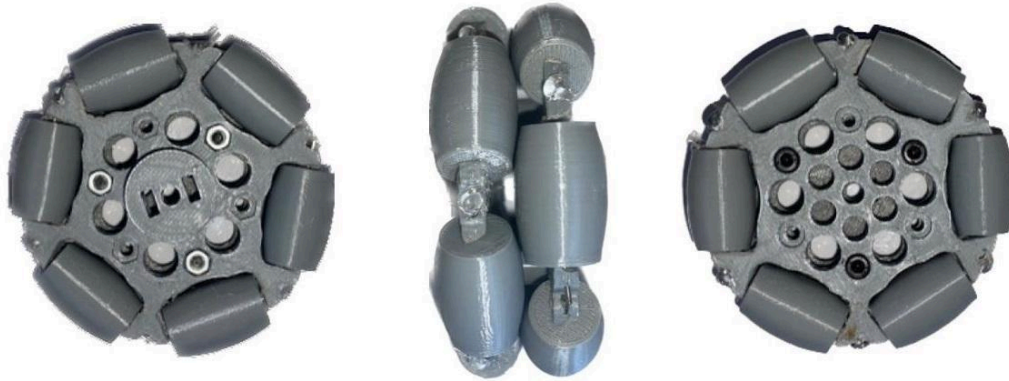


Figure 4.1: Omnidirectional Wheel after Assembling

4.2.2 Cell Implementation

We determined the cell dimensions and designed how to optimally distribute the wheels on the cell then designed and implemented the complete cells board as Shown in Figure 4.2 . The board consists of 4 cells; each cell contains 3 omnidirectional wheels with a total of 12 wheels

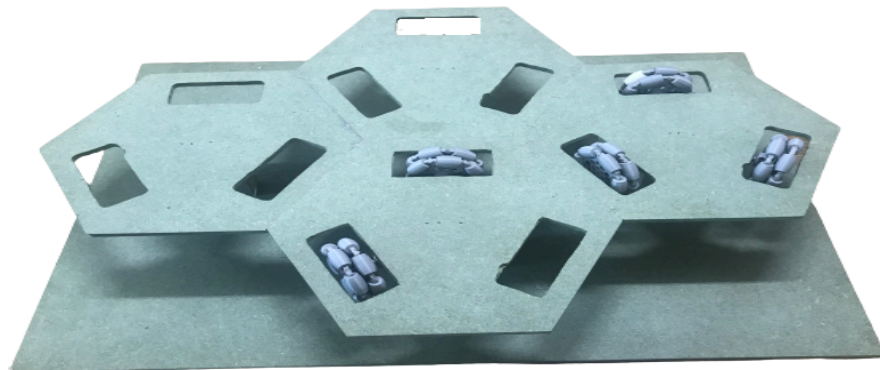


Figure 4.2: Cells Board

4.2.3 DC Motor Connection

To connect the DC motors with Raspberry Pi using the L298N motor driver as shown in Figure 4.3, we follow these steps:

1. Connect the power supply to the motor driver. Also, connect the ground (GND) of the power supply to the ESP32 ground pin.
2. Connect the motor driver's input pins (IN1, IN2) to different GPIO pins on the ESP32. These pins will control the direction of the motors (forward or backward).
3. Connect the motor driver's output pins (OUT1, OUT2) to the corresponding terminals of the Dc motors.
4. Finally, we write a Python program, to control the motors through the ESP32, and use the GPIO library to set the appropriate GPIO pins to High or Low to control the motor direction.

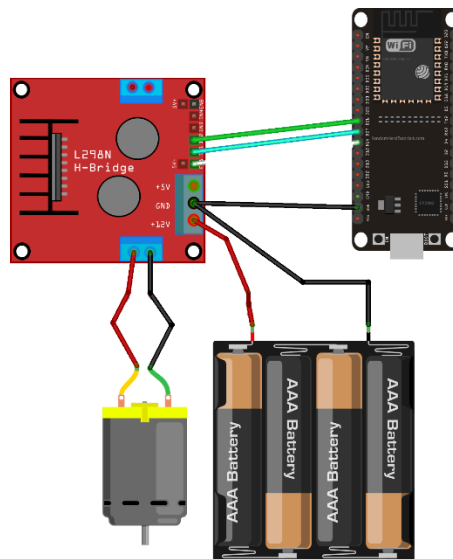


Figure 4.3: DC Motor Connected with Raspberry Pi

4.2.4 Load Cell Connection

To measure the weight of the package we made a connection between the load cell and Raspberry Pi using the XH711 module sensor as shown in Figure 4.4.

- load cell interface to hx711:

Red line: E+ pin
Black line: E- pin
Green line: A- pin
White line: A+ pin

- hx711 interface to RPI

VCC to Raspberry Pi Pin 2 (5V)
GND to Raspberry Pi Pin 6 (GND)
DT to Raspberry Pi Pin 29 (GPIO 5)
SCK to Raspberry Pi Pin 31 (GPIO 6)

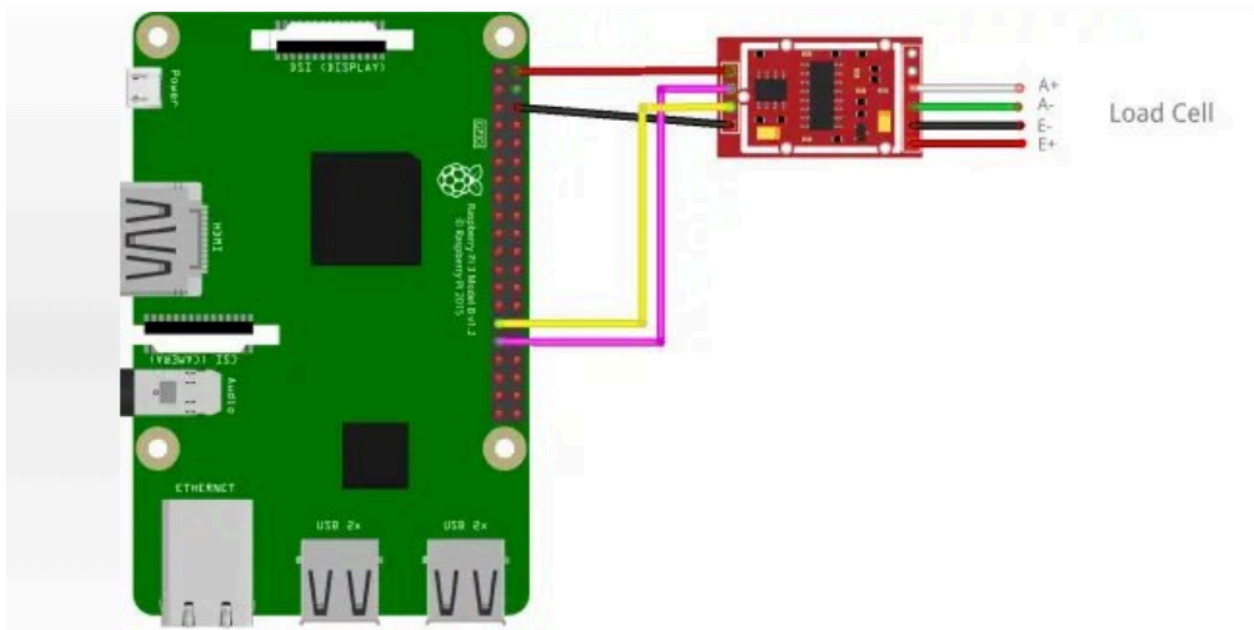


Figure 4.4: load Cell Connection with Raspberry Pi

4.2.5 Path Planning Sketch

We sketch and design the path planning for the movement of the package from the entry point to the exit point and how to control the motion of the wheels required to deliver the package to the desired sorting area.

The path from an entry point to the unsorted area is shown in Figure 4.5.

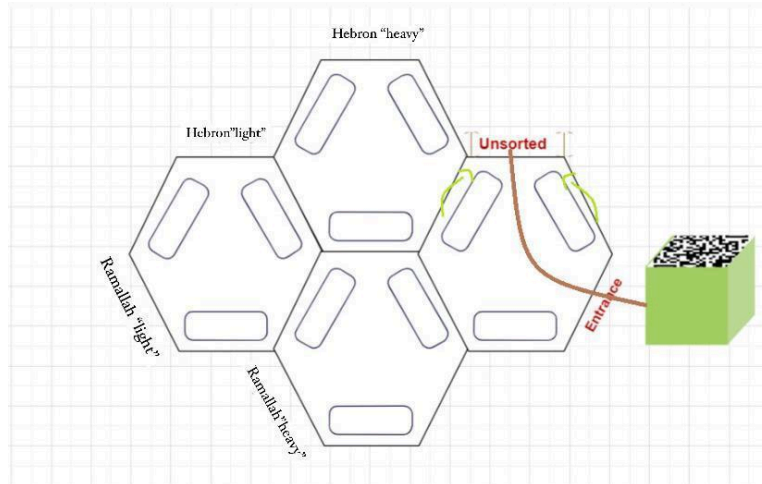


Figure 4.5: The Path for The Unsorted Area

The path from an entry point to the Hebron light area is shown in Figure 4.6.

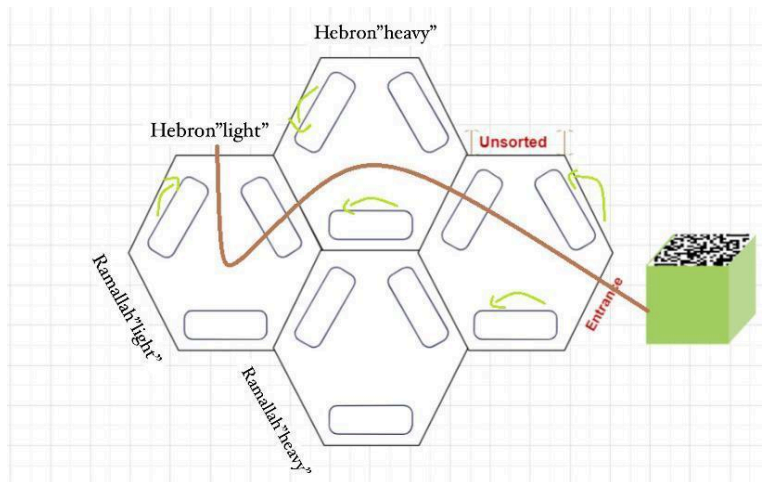


Figure 4.6: The Path for The Hebron Light Area

The path from an entry point to the Hebron heavy area is shown in Figure 4.7.

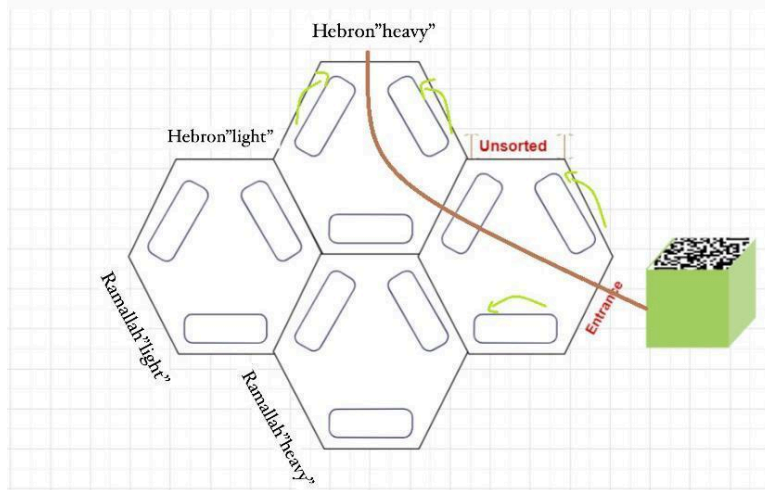


Figure 4.7: The Path for The Hebron Heavy Area

The path from an entry point to the Ramallah light area is shown in Figure 4.8.

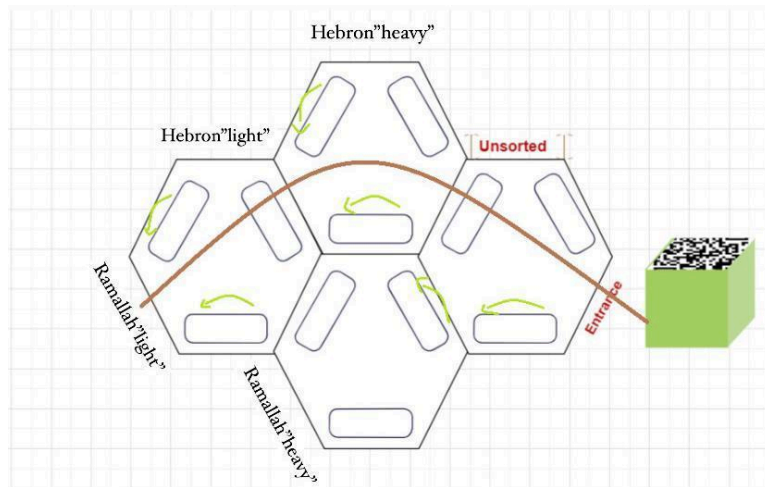


Figure 4.8: The Path for The Ramallah Light Area

The path from an entry point to the Ramallah heavy area is shown in Figure 4.9.

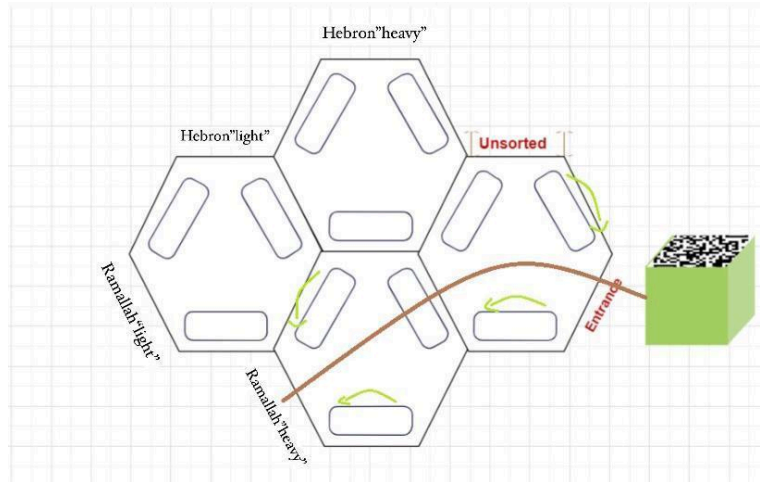


Figure 4.9: The Path for The Ramallah Heavy Area

4.2.6 Download Operating System in Raspberry Pi

We downloaded a pre-installed copy of the Raspbian operating system from the raspberry pi.org website, we installed that disk image in the Raspberry Pi SD card. After inserting that SD card in the Raspberry Pi and connecting it to the power supply, then the Raspbian operating system was ready to be used to build our system.

4.3 Library installation

In order for the system to start working, we need several libraries to be installed on the OS of the raspberry pi. This required writing a Raspbian OS image to an SD card. The image we used had every necessary library preinstalled, including OpenCV, ZBar, HX711, and Python's version.

4.4 Software Implementation

4.4.1 OpenCV Setup

We need to install the OpenCV library to capture a frame for the QR code from the video stream and read this QR. To install it we will enter the following commands into the Raspberry Pi terminal:

```
Sudo apt-get update  
Sudo apt-get install python3-opencv  
Pip3 install opencv-contrib-python
```

4.4.2 ZBar Setup

We need to install the ZBar library to decode the QR code and convert it to the respective information. To install it we will enter the following commands into the Raspberry Pi terminal:

```
[REDACTED]
```

4.4.3 HX711 Setup

This library allows driving an HX711 load cell amplifier with a Raspberry Pi. To install it we will enter the following command into the Raspberry Pi terminal:

```
[REDACTED]
```

before downloading this library, we need to download RPI.GPIO Package to enable GPIO pins for the Raspberry Pi using the below command:

```
[REDACTED]
```

4.4.4 QR Code Generation

We designed a QR code for Hebron City shown in Figure 4.10 and a QR code for Ramallah City shown in Figure 4.11 using the free QR code generation online program.



Figure 4.10: QR Code for Hebron



Figure 4.11: QR Code for Ramallah

4.5 Implementation Issues and Challenges

We faced a challenge during the system implementation: the lack of a sufficient number of omnidirectional wheels and they very expensive price. Therefore, we turned to the solution of printing the wheels in 3D printing, and that was a great challenge for us, as there was a possibility that printing would not succeed and the wheels would not move smoothly and in the required shape. After we tested and sure from printing one wheel we faced a challenge in the time printing the wheels, and will we be able to finish them and complete the rest of the project during the semester or not. Table 4.2 below illustrates the name of the pieces needed for one wheel, the time for printing these pieces, and the amount of filament material that the one wheel needs, as shown in the table one wheel

needs 5 hours and 18 min time, and 14.28-meter filament, so we had to reduce the number of cells and the number of wheels, and limit the system to the cities of Hebron and Ramallah only to finish the project in the required time. The total time for printing all wheels is 64 hours and the amount of material used for all wheels is 171 meters.

Name	Number (NO)	Time (T)	Amount of filament (A)	Printing Time	Length of Material (meter)
Omni_2stl	1 pcs	1 hour 19 min	2.75 meter	1 hour 19 min	2.75 meter
Omni_1_5mm_hole.stl	1 pcs	1 hour 27 min	3.19 meter	1 hour 27 min	3.19 meter
Omni_roller.stl	12 pcs	15 min	0.65 meter	3 hours	7.8 meter
Omni_spacer.stl	3 pcs	6 min	0.18 meter	18 min	0.54 meter
Total				5 hours 18 min	14.28 meter

Table 4.2: Total Time and Filament Need to Print One Wheel

Chapter 5

Result and Validation

5.1 Overview

In this chapter, we will discuss the testing of all components of the system and the results obtained.

5.2 Hardware Testing

This section discusses the testing process of each of our hardware components.

5.2.1 Testing Raspberry Pi

We have tested all points and GPIO pins of the Raspberry Pi and ensured that they are working without any problem.

5.2.2 Testing Raspberry Pi Camera

We connected the camera on the Raspberry using the MIPI CSI interface (the port in the center near the USB ports) as shown in Figure 5.1. Then we enabled the Camera Interface from the Raspberry Pi configuration screen as shown in Figure 5.2, and ran the below commands to test it by taking an image and recording a video.

```
$ raspistill -o test.jpeg
```

```
$ raspivid -o test.h264 -t 30000
```

It captured the photo and a video.

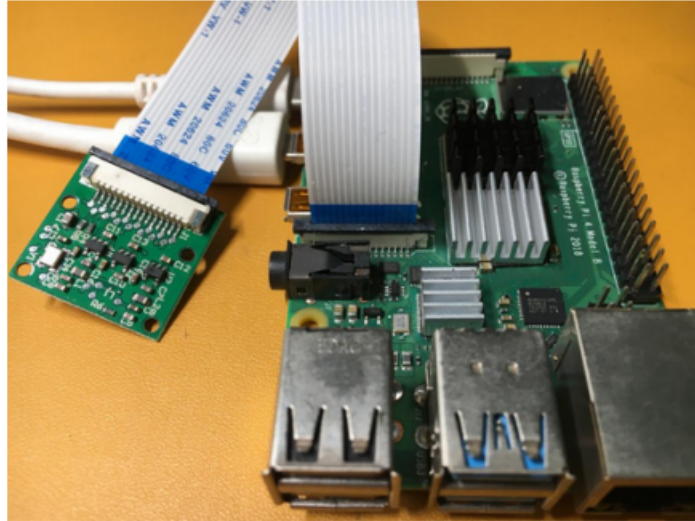


Figure 5.1: Camera Connected with Raspberry Pi

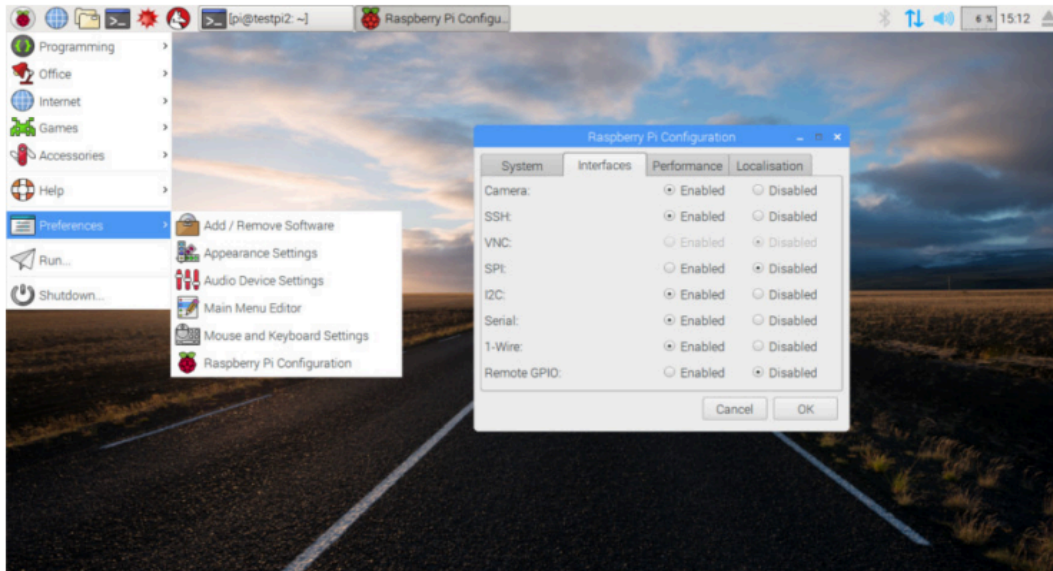


Figure 5.2: Enabling The Camera Interface from Raspberry Pi Configuration

5.2.3 Testing load cell

To Test whether the load cell is working correctly, the first step is to connect the load cell pins with the load cell amplifier (HX711) and other pins of the HX711 with Raspberry Pi, then we run a Python code to measure the weight after that we test two cases one case when the weight is light the result of it is shown in Figure 5.3, and another case when the weight is heavy the result of it is shown in Figure 5.4.

3.1212389380530974
light weight

Figure 5.3: The Result for Measuring 3 gm

839.3816371681415
heavy weight

Figure 5.4: The Result for Measuring 830 gm.

As shown in the result, there is a very small percentage of error in the measurement due to the inaccuracy of the load cell version.

5.2.4 Testing Dc motor and Wheel

DC motor was tested by connecting with the power supply.

The wheel was tested by connecting with a DC motor, then a Python code was written to be run.

5.3 Software Testing

5.3.1 Testing QR reader

To test and read the QR code we connect the Raspberry Pi camera with the raspberry pi and enable the camera interface, at the first the OpenCV library captures a frame from the video stream and processes it, once the OpenCV captures a frame then we pass it to ZBar library to decode the QR code and convert it to respective information.

The result of testing the QR code for Hebron City is shown in Figure 5.3.

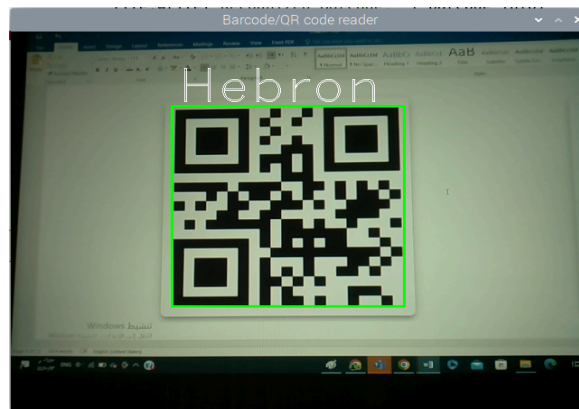


Figure 5.3: Scan the QR code for Hebron

The result of testing the QR code for Ramallah City is shown in Figure 5.4.

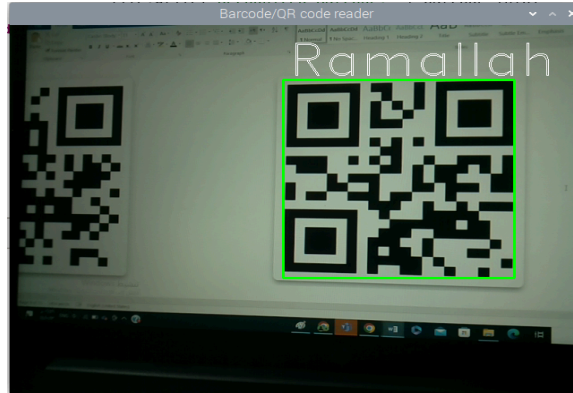


Figure 5.4: Scan the QR code for Ramallah

5.4 Integration Testing

After ensuring that all the components tested are working well, we started assembling and integrating the parts with each other to make the system ready to work.

5.5 Testing Scenario and Result

Scenario #	QR Code	Weight	Result
1	No QR code on the package	Or The weight of the package above or less than 1 kg	The system directed the package to an unsorted area through the specified path.
2	Hebron	0.5 kg	The system directed the package to the Hebron "light" area through the specified path.
3	Hebron	1 kg	The system directed the package to the Hebron "heavy" area through the specified path.
4	Ramallah	0.5 kg	The system directed the package to the Ramallah "light" area through the specified path.
5	Ramallah	1 kg	The system directed the package to the Ramallah "heavy" area through the specified path.

Table 5.1: Testing Scenario and Result

Chapter 6

Conclusion and Future Work

6.1 Overview

This chapter shows the project conclusion, and the main achieved objectives, and will propose future works that can be developed on the system.

6.2 Conclusion

In conclusion, the Omnidirectional Sorting Conveyor project is a groundbreaking solution for efficient package sorting. It utilizes QR codes and weight measurements to accurately sort packages, optimizing time and resources while minimizing errors. This innovative project showcases the potential of advanced technology in revolutionizing logistics practices, setting a new standard for package sorting systems. Its implementation promises improved efficiency, reduced errors, and enhanced customer satisfaction, highlighting the power of innovation and technology in shaping a more seamless and interconnected world.

6.3 Future Work

In the future, we look forward to improving the omnidirectional sorting conveyer, some of the future work are suggested:

1. Developing the system to include all cities of the state of Palestine.
2. Developing an electronic arm that pushes the package to the conveyer instead of manual pushing.
3. As recommended, convert the system from open-loop to closed-loop control.

References

[1] Admin, G.T.F.R. (2021) *What is Omni Wheel and how does it work?* GTFRobots. Available at: <https://gtfrobits.com/what-is-omni-wheel/> (Accessed: November 3, 2022).

[2] Faginey Omni Wheel for robot, Robotic Omni Directional Wheel, 48mm OMNI directional wheel loading 2kg for robotic competition robotic toy parts (no date) Walmart.com. Available at: <https://www.walmart.com/ip/FAGINEY-Omni-Wheel-For-Robot-Robotic-Omni-Directional-Wheel-48mm-Omni-Directional-Wheel-Loading-2kg-For-Robotic-Competition-Robotic-Toy-Parts/189231461> (Accessed: November 3, 2022).

[3] Utmel (2020) What are weight sensors? Utmel. Utmel Electronics. Available at: <https://www.utmel.com/blog/categories/sensors/what-are-weight-sensors> (Accessed: November 3, 2022).

[4] 5KG weight sensor philippines - makerlab electronics (no date) Makerlab Electronics - Bits and pieces of Hardware and Electronic Stuffs! Available at: <https://www.makerlab-electronics.com/product/5kg-weight-sensor/> (Accessed: November 3, 2022).

[5] Who Invented the QR Code? QR Code History & Creator (2020) Who invented the QR code? QR code history & creator. Scott Schulfer. Available at: <https://www.sproutqr.com/blog/qr-code-history#:~:text=The%20QR%20code%20was%20invented,They%20invented%20the%20QR%20code.> (Accessed: November 3, 2022).

[6] Schulfe, S. (2020) How Do QR Codes Work? QR Code Technical Basics, How do QR codes work? QR code technical basics. Sprout QR. Available at: <https://www.sproutqr.com/blog/how-do-qr-codes-work> (Accessed: November 3, 2022).

[7] Wainwright, C. (2022) How to make a QR code in 5 easy steps, HubSpot Blog. HubSpot. Available at:

<https://blog.hubspot.com/blog/tabid/6307/bid/29449/how-to-create-a-qr-code-in-4-quick-steps.aspx> (Accessed: November 3, 2022).

[8] Abu Khalaf, Islam and Abu Arafah, Murad (2018) Design and Implementation of an Automatic Sorting Machine, PPU DSpace Repository. Palestine Polytechnic university. Available at: <http://localhost:8080/xmlui/handle/123456789/6307> (Accessed: November 3, 2022).

[9] Msulam, Diaan and AL-Hraie, Anas (2019) Potato Sorting Machine Based on LabVIEW with Image Processing strategy, PPU DSpace Repository. Palestine polytechnic university. Available at: <https://scholar.ppu.edu/handle/123456789/7060> (Accessed: November 3, 2022).

[10] ijarse (2018) Development of an Automatic Sorting Machine based on product colour and weight characteristics with Conveyor Belt and sensors. International Journal of advanced Research in Science and Engineering. Available at: http://www.ijarse.com/images/fullpdf/1523788273_PG509ijarse.pdf (Accessed: November 3, 2022).

[11] Sujit and Miyani, J. (2022) Automatic sorting machine using conveyor, Engineering Arena. Available at: <https://www.enggarena.net/automatic-sorting-machine-using-conveyor/> (Accessed: November 3, 2022).

[12] RasPiTV, alex eames – et al. (2021) *Raspberry pi 4 on sale now from \$35, Raspberry Pi*. Available at: <https://www.raspberrypi.com/news/raspberry-pi-4-on-sale-now-from-35/> (Accessed: November 13, 2022).

[13] Admin (2021) *Object sorting using Raspberry Pi, Electronics workshop*. Available at: <https://electronicsworkshops.com/2021/09/14/object-sorting-using-raspberry-pi/> (Accessed: November 13, 2022).

[14] *Raspberry pi camera module v2.1* (no date) *The Pi Hut*. Available at: <https://thepihut.com/products/raspberry-pi-camera-module> (Accessed: November 13, 2022).

[15] Administrator (2021) *DC motor types - brushed, brushless and DC Servo Motor, Electronics Hub*. Available at: <https://www.electronicshub.org/dc-motor/> (Accessed: November 13, 2022).

[16] A. Abed and S. Rahman, "Python-based raspberry pi for hand gesture recognition," International Journal of Computer Applications, vol. 173, pp. 975-988, 09 2017.

[17] Qengineering. (n.d.). *QR_SCANNER_RASPBERRY_PI/readme.md at main · qengineering/QR_SCANNER_RASPBERRY_PI*. GitHub. Retrieved December 25, 2022, from https://github.com/Qengineering/QR_scanner_Raspberry_Pi/blob/main/README.md

[18] Tinker Board. (n.d.). Retrieved November 24, 2022, from <https://tinker-board.asus.com/product/tinker-board.html>

[19] Alderson, A. (2021, September 28). *Asus releases two new Tinker Board sbcs to take on the Raspberry Pi*. Notebookcheck. Retrieved November 24, 2022, from <https://www.notebookcheck.net/ASUS-releases-two-new-Tinker-Board-SBCs-to-take-on-the-Raspberry-Pi.505391.0.html>

[20] B. Earl, "All about stepper motors," May 2014, accessed on 11.11.2021. [Online].

Available: <https://learn.adafruit.com/all-about-stepper-motors>

[21] "series stepper motors," accessed on 11.11.2021. [Online]. Available: <http://en.cwmotor.cn/product/37.html>

[//en.cwmotor.cn/product/37.html](http://en.cwmotor.cn/product/37.html)

[22] Google. (n.d.). Google search. Retrieved December 6, 2022, from https://www.google.com/search?q=omnidirectional%2Bwheels%2Ball%2Bdirection&rlz=1C1GCEU_enPS993PS993&source=lnms&tbm=isch&sa=X&ved=2ahUKEwjz473x3eT7AhUF2qQKHTZgBbgQ_AUoAXoECAIQAw&biw=1229&bih=531&dpr=1.56#imgrc=IZFe1vGr05rodM

[23] Nagrath, A. (2022, September 2). *Mecanum wheels! Astonishing Engineering of Wheels*. Engineering Exploration. Retrieved November 24, 2022, from <https://engineeringexploration.com/mecanum-wheels-astonishing-engineering-of-wheels/>

[24] File: *mecanum wheel control principle.svg*. Wikimedia Commons. (n.d.). Retrieved November 24, 2022, from https://commons.m.wikimedia.org/wiki/File:Mecanum_wheel_control_principle.svg

[25] McManus, S., & Cook, M. (2021). *Raspberry Pi*. Amazon. Retrieved December 6, 2022, from <https://www.amazon.com/raspberry-pi/s?k=raspberry%2Bpi>

[26] Amazon.com: *SmartFly Info Tinker Board 2s 6-core 2.0 ghz rockchip ...* (n.d.). Retrieved December 6, 2022, from <https://www.amazon.com/SmartFly-info-Bluetooth-Connectivity-Dual-Display/dp/B08V51FHB8>

[27] *What is a webcam? - definition from Techopedia*. Techopedia.com. (n.d.). Retrieved December 6, 2022, from <https://www.techopedia.com/definition/5333/webcam>

[28] Google. (n.d.). Google. Retrieved December 6, 2022, from https://www.google.com/search?q=definition%2Bof%2Bweb%2Bcamera&rlz=1C1GCEU_enPS993PS993&hl=ar&source=lnms&tbm=isch&sa=X&ved=2ahUKEwiujpeF-eT7AhUFSuUKHTRABIkQ_AUoAXoECAMQAw&biw=1229&bih=577&dpr=1.56#imgrc=phxraeHYqp6HMM

[29] Google. (n.d.). Google search. Retrieved December 6, 2022, from https://www.google.com/search?q=omnidirectional%2Bwheels&rlz=1C1GCEU_enPS993PS993&source=lnms&tbm=isch&sa=X&ved=2ahUKEwiCzLXr7uT7AhWugP0HHQIHB-wQ_AUoAXoECAIQAw&biw=1229&bih=531&dpr=1.56#imgrc=39c5cCdy-KQ4cM

[30] Google. (n.d.). Google search. Retrieved December 6, 2022, from https://www.google.com/search?q=omnidirectional%2Bwheels&rlz=1C1GCEU_enPS993PS993&source=lnms&tbm=isch&sa=X&ved=2ahUKEwiCzLXr7uT7AhWugP0HHQIHB-wQ_AUoAXoECAIQAw&biw=1229&bih=531&dpr=1.56#imgrc=39c5cCdy-KQ4cM

3PS993&source=Inms&tbm=isch&sa=X&ved=2ahUKEwiCzLXr7uT7AhWugP0HHQIHB-wQ_AUoAXoECAIQAw&biw=1229&bih=531&dpr=1.56#imgrc=5MHftJ2tsaUWfM

[31] *Product. EK-i/EW-i Series Compact Balances | Balances | Weighing | Products | A&D.* (n.d.). Retrieved December 6, 2022, from https://www.aandd.jp/products/weighing/balance/compact/ekwi_series.html

[32] *Micro load cell (weight sensor) with 50kg capacity.* Hatchnhack Cart. (n.d.). Retrieved December 6, 2022, from <https://www.hnhcart.com/products/micro-load-cell-weight-sensor-with-50kg-capacity>

[33] *EK-300i compact balance, 300G x 0.01g with external calibration.* A&D Weighing. (n.d.). Retrieved December 6, 2022, from <https://weighing.andonline.com/product/everest-ewek-series-compact-balances/ek-300i>