Department of Computer Science & Engineering
The University of Texas at Arlington

Architectural Design Specification
CSE 4316: Senior Design I
Fall 2023

Robo Crew
RV8 Workcell

Akshay Paluri
Ameen Mahouch
Muhammad Anas
Hyun Ho Kim
Kundan Singh Mahato

Revision History

Revision	Date	Author(s)	Description
0.1	11.06.2023	AM, MA, AP, HK, KM	document creation

Contents

List of Figures

List of Tables

1 Introduction

This product is a high level working prototype of an industrial level robot which serves the purpose of ordering and arranging the boxes by scanning the QR codes. The boxes will be picked by a gripper (custom designed) and will be sorted. The robot arm is connected to a PLC controller and the arm is set on to an additional axis (i.e. linear rail) which is also connected to a servo amplifier present in the PLC. This PLC controller is connected to a PC containing various software to control the robot arm. Raspberry Pi is used for the QR code detection and decision making which is connected to a camera.

2 System Overview

This section outlines the architectural strategy for the flow of the RV8 work cell system, defining the top-level logical view of the design. The system is structured into three distinct layers: Input, Processing, and Output. Each of these layers serves a specific and vital function within the system, enabling the robot to perform dynamic tasks based on the processing of sensor data. This section includes a high-level block diagram that visually illustrates the relationships and interactions between these layers, providing a comprehensive overview of the system's architecture.

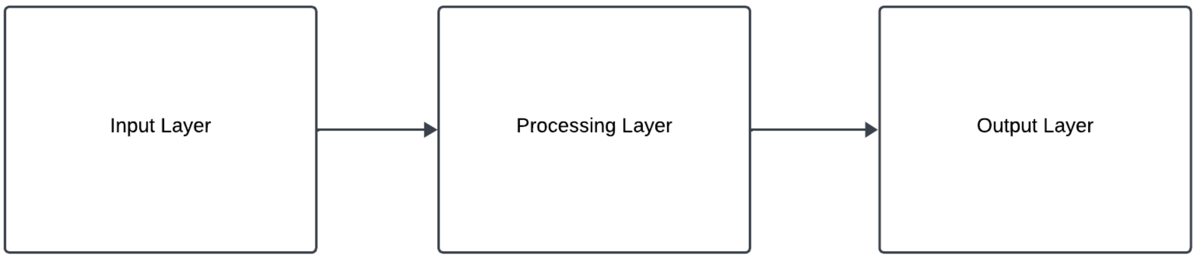

Figure 1: An overview of the higher level layers.

2.1 Input Layer Description

Since the robot arm will perform tasks as desired by users, this input layer will play an important role in completing these tasks. These sensors gather information from external sources, and based on this information, the internal configuration will be adjusted to respond to different situations. In the input layer, we will have several sensor components, including a QR scanner, a Gate sensor, and E-stops. These sensors will communicate with the processing layer, which primarily controls the robot. The QR scanner reads QR codes and sends the data in digital format to the controller through the Raspberry Pi. The E-stops send a 'True' signal to the controller when any of the E-stops are pressed, causing an immediate system halt, regardless of the ongoing process. Lastly, the Gate sensor determines whether the cage gate is open or closed. When the gate is open, the robot arm will move slowly, as it controls the robot arm's speed and communicates with the controller

2.2 Processing Layer Description

Processing layer plays very important role in this product. All the decision making and handling is done in this layer. This layer encompass PLC controller, raspberry pi and a PC. These are interconnected to each other and perform necessary communication to perform given task accurately. PLC controller's primary function is to control robot arm (all joints) and additional axis (in this case a linear rail). Where as, the function of Raspberry pi is to collect data and send it to PC where the decision is made, whether to perform the task on the incoming object (box in this case). PC and PLC talk to each other, in order to align all the joints of the arm and perform task.

2.3 Output Layer Description

The output Layer comprises two key components: the robot arm and indicator subsystems, both vital for efficient box palletizing and depalletizing operations. The robot arm, equipped with a gripper, linear rail, and joints, is responsible for handling boxes, creating and removing grips, and configuring its position for tasks. On the other hand, the indicator subsystem employs an industrial light tower to provide visual feedback to operators, signaling different operating modes and safety alerts. These subsystems work in tandem to ensure the smooth execution of tasks within the RV8 work cell system.

3 Subsystem Definitions & Data Flow

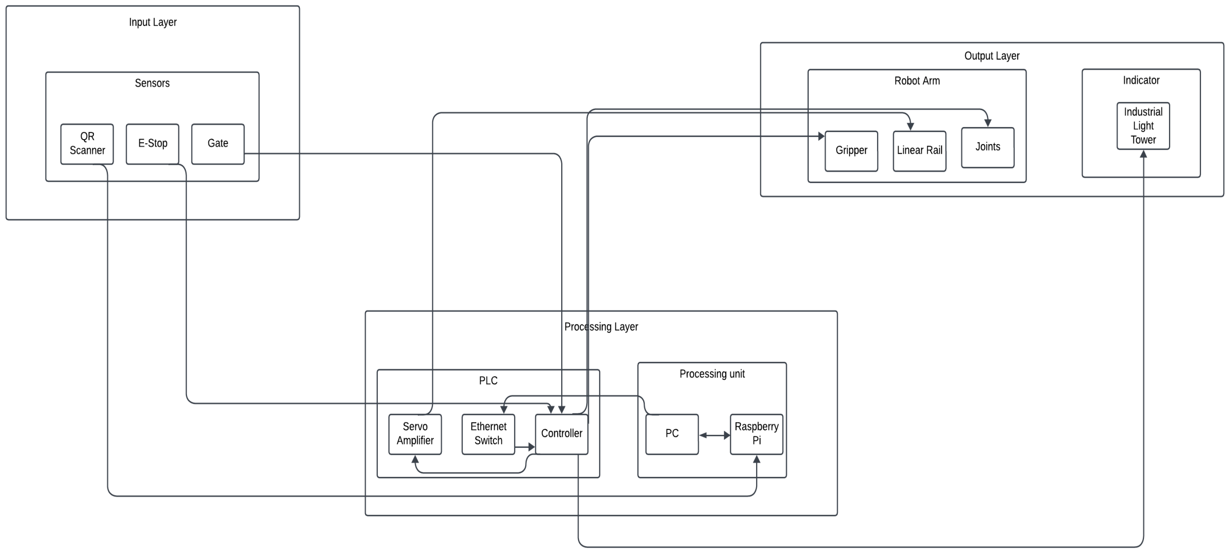

Figure 2: Data flow diagram of workcell.

4 Input Layer Subsystems

The input layer comprises various sensors, such as the QR scanner, E-Stop, and gate sensors. Programmable Logic Controllers (PLCs) or Raspberry Pi can receive input signals by reading data from external sources through these sensors in the environment. We can configure the robot arm and the linear rail to execute a task, and in case of an emergency, the E-Stop can halt our program instantly.

4.1 QR Scanner

The QR Scanner functions as a vision sensor, capturing images and converting them into digital signals. In our project, QR codes will include location information, enabling our robot arm to accurately position boxes at designated locations.

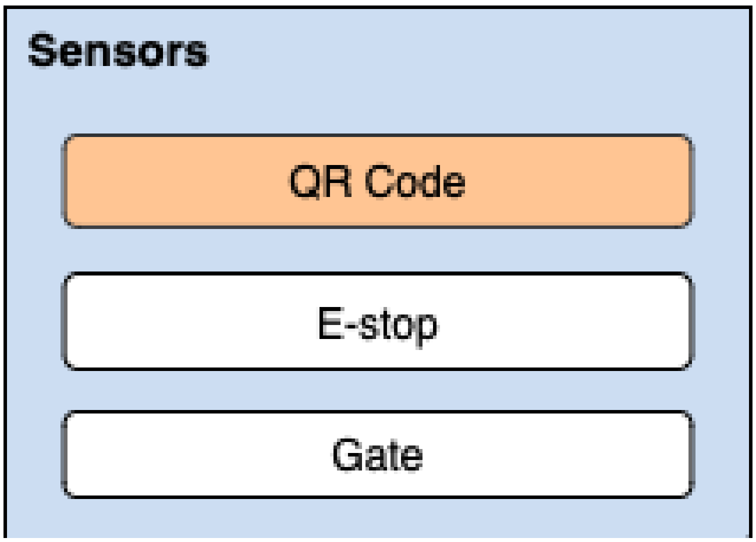

Figure 3: QR code subsystem description diagram

4.1.1 Assumptions

The QR scanner will interface with the Raspberry Pi. It will transmit the location data obtained from the scanner and any commands from our PC to control the robot arm. If, for any reason, the QR scanner is unable to read the QR code, such as due to low lighting conditions, it will not transmit any digital data to the Raspberry Pi.

4.1.2 Responsibilities

The QR scanner, mounted on the robot arm, reads QR codes on boxes. Each QR code contains location information indicating where the box should be positioned. The QR scanner is connected to a Raspberry Pi to transmit the digital data.

4.1.3 Subsystem Interfaces

Table 2: Subsystem interfaces

ID	Description	Inputs	Outputs
#01	read QR code and send the data	image	digital data

4.2 E-Stop

E-Stops serve as safety features, allowing users to halt the robot in the event of unexpected actions or when they desire. This sensor can stop the program, regardless of the current processing state.

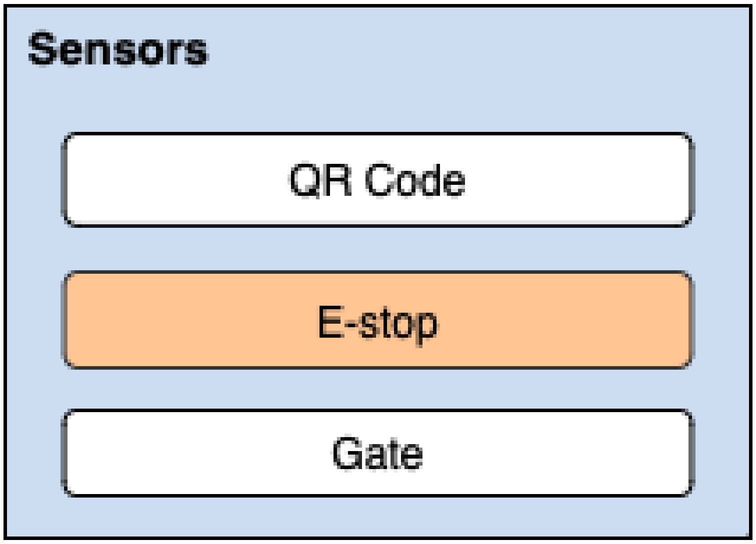

Figure 4: E-stop subsystem description diagram

4.2.1 Assumptions

When any of the E-stop buttons is pressed, the input value changes, and this value is then relayed to the controller to halt the system. E-stops can be pressed to immediately stop the robot in case of potential collisions, unexpected behavior, or excessive temperature increases.

4.2.2 Responsibilities

If users press any of the E-stops, the system stops by transmitting the changed input value to the controller.

4.2.3 Subsystem Interfaces

Table 3: Subsystem interfaces

ID	Description	Inputs	Outputs
#02	E-stop signal	button status	0 or not 0

4.3 Gate

The gate sensor is a safety component that detects whether the gate is open or closed.

4.3.1 Assumptions

When the gate is open, the gate sensor sends a 'True' signal to the control, causing the robot to move slowly.

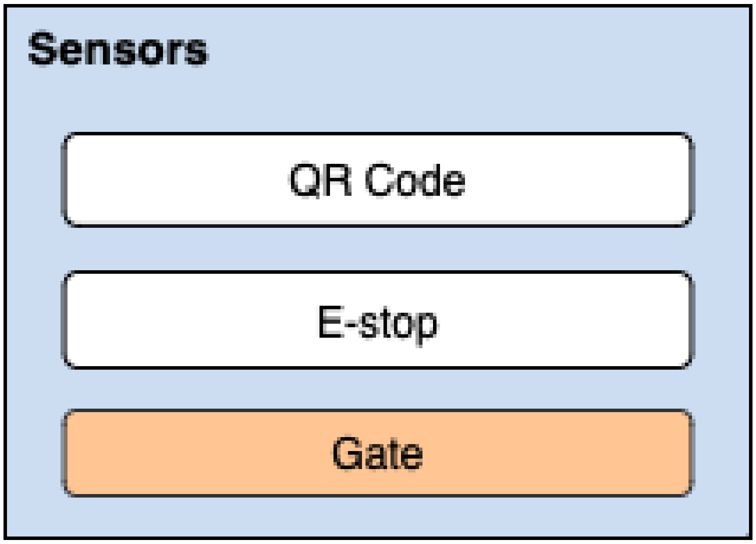

Figure 5: Gate subsystem description diagram

4.3.2 Responsibilities

Since the gate sensor detects value changes, it must transmit the changed value to the controller for a response.

4.3.3 Subsystem Interfaces

Table 4: Subsystem interfaces

ID	Description	Inputs	Outputs
#03	Gate open or close	gate	True or False

5 Processing Layer Subsystems

5.1 PLC

This section should be a general description of a particular subsystem for the given layer. For most subsystems, an extract of the architectural block diagram with data flows is useful. This should consist of the subsystem being described and those subsystems with which it communicates.

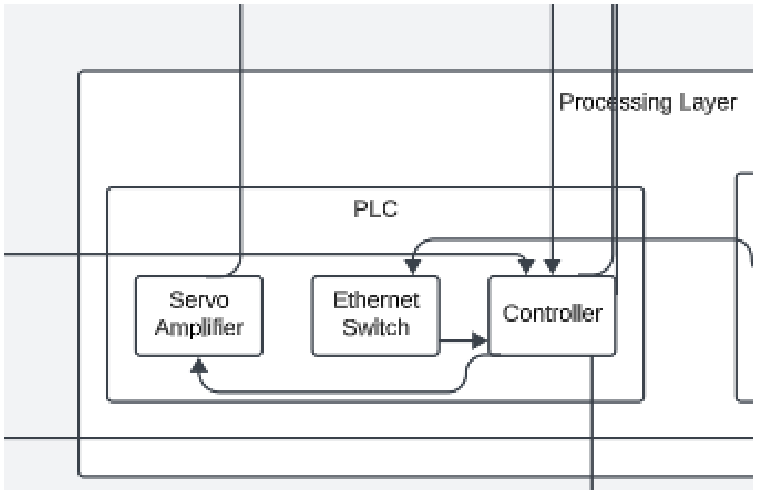

Figure 6: PLC subsystem.

5.1.1 Assumptions

PLC is like a central processing unit of this product. The input comes from the input layer. Then, the input is processed and sent to the output layer.

5.1.2 Responsibilities

PLC consists of many components. But the three major components are the Servo amplifier, Ethernet switch, and Controller. Servo amplifier: The Servo amplifier's job is to move the linear rail into the appropriate position so that the arm can perform its targeted operation (picking box). The amplifier sends a signal from the Controller because it is connected as an additional axis to the controller beside other joints of the robot arm. Ethernet Switch: The ethernet switch is present inside the controller. Since this robot is being controlled via TCP/IP, it is very important to have a stable wired connection. This switch makes communication possible between the PC and the Controller. Controller: The controller, as the name sounds, controls the robot arm and rail. The instructions or commands it takes as input from the PC result in the movement of the robot arm accordingly. The controller also generates output to other parts of the product. It sends signals continuously to the indicator, indicating the robot status. Green is working correctly. Orange attention is required. Red stopped working. It instructs the arm when to align all the joints and pick the box using a gripper.

5.1.3 Subsystem Interfaces

Table 5: Subsystem interfaces

ID	Description	Inputs	Outputs
#1	Takes input from controller	Controller	Movement of Linear Rail
#2	Takes input from PC	PC	Send signal to Controller
#3	Takes input from different components	a. Input from PC via ethernet switch b. Input from gate sensor c. Input from E-Stops	a) Sends the signal to servo amplifier and align joints to perform action. b) If the gate sensor is triggered, arm operation is terminated c) If e-stops are hit, arm operations are terminated

In this section, the processing unit layer is described in some detail in terms of its specific subsystems. The processing unit comprises two primary components: the host PC and the Raspberry Pi. This layer plays a crucial role in the robotic system by acquiring input data from sensors and processing it before transmitting the processed data to the robot controllers.

5.2 Host PC

The host PC acts as a powerful processing unit for in-depth analysis, decision making, and control. It receives pre-processed data from the Raspberry Pi, including data from the QR code, and it performs trajectory planning and higher level decision making.

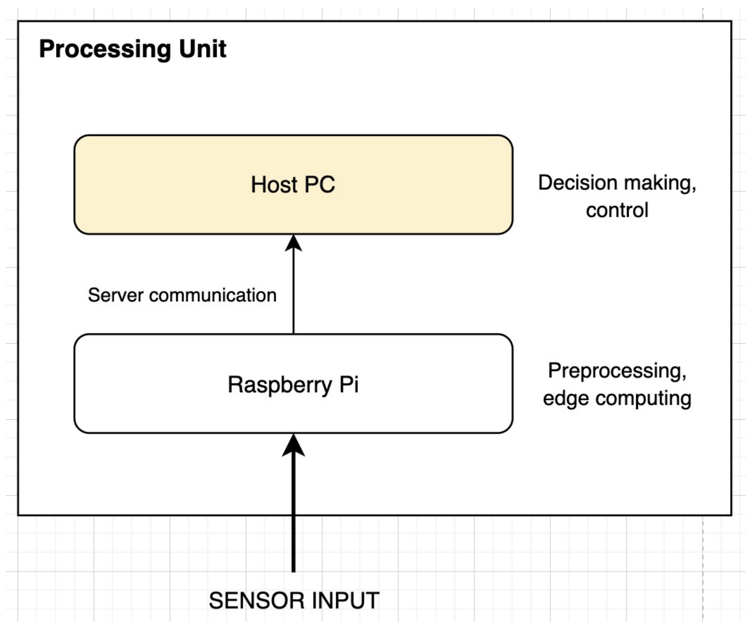

Figure 7: Host PC subsystem

5.2.1 Assumptions

- The host PC assumes a stable connection to the Raspberry Pi for data transfer.
- The host PC assumes that data received from Raspberry Pi is processed correctly.

5.2.2 Responsibilities

- Data Analysis: The host PC is responsible for analyzing the data received from the Raspberry Pi, which includes processing sensor data, decoding QR codes, and running various algorithms for object recognition.
- Decision-Making: Based on the data analysis, the host PC is responsible for making high-level decisions, such as robot control commands, path planning, and task scheduling.
- Communication with PLC: The host PC is responsible for communicating with the Programmable Logic Controller (PLC) to issue control commands and receive status updates.
- Output to Indicators: The host PC is responsible for sending instructions to indicators to provide visual feedback to operators.

5.2.3 Subsystem Interfaces

Table 6: Subsystem interfaces

ID	Description	Inputs	Outputs
#01	Data input from Raspberry Pi	Preprocessed sensor data QR code results	Control instructions
#02	Communication with PLC	N/A	Control commands

5.3 Raspberry Pi

The Raspberry Pi serves as an integral component of the processing unit within the robot system. It fulfills the role of an edge computing device, responsible for initial data processing and interfacing between the sensors and the host PC. Specifically, the Raspberry Pi is responsible for collecting data from a variety of sensors, including those for QR code scanning and other environmental monitoring.

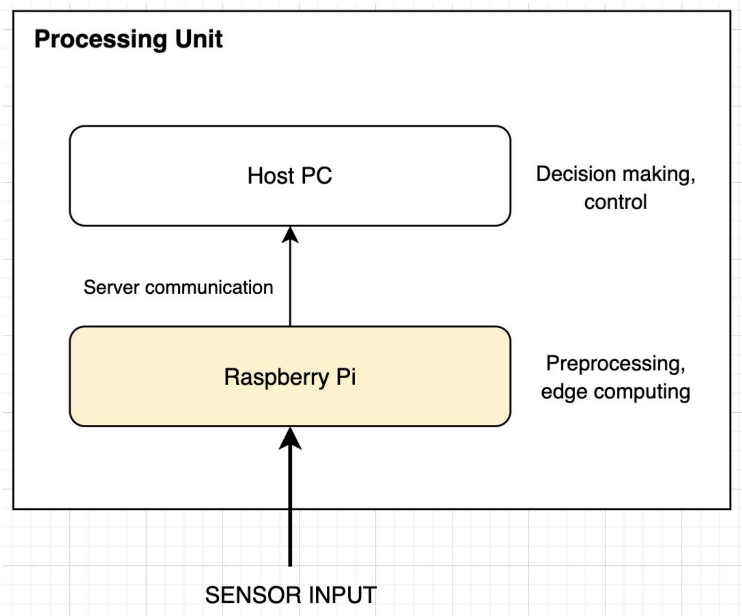

Figure 8: Raspberry Pi subsystem

5.3.1 Assumptions

- The Raspberry Pi is properly installed on the robot arm and receives sensor input data correctly.
- The Raspberry Pi receives data in the expected format.
- The Raspberry Pi has the necessary software and libraries installed for data preprocessing.
- The Raspberry Pi has an established connection with the Raspberry Pi.

5.3.2 Responsibilities

- Sensor Data Collection: The Raspberry Pi is responsible for collecting data from various sensors, such as cameras and QR code scanners.
- Data Preprocessing: The Raspberry Pi preprocesses the collected data, which may include image processing, decoding QR codes, and initial data filtering.
- Data Transfer: It is responsible for transferring the preprocessed data to the host PC for further analysis and decision-making.

5.3.3 Subsystem Interfaces

Table 7: Subsystem interfaces

ID	Description	Inputs	Outputs
#01	Sensor data	Sensor data streams	Preprocessed data
#02	Communication with host PC	N/A	Data to be analyzed

6 Robot Arm Layer Subsystems

This section talks about the layer of the robot arm and how each components are going to interact with necessary data inputs to perform the task of palletizing and depalletizing the boxes. The primary goal of this layer is to interact with the boxes and configure the necessary position the arm needs to be in to execute the task.

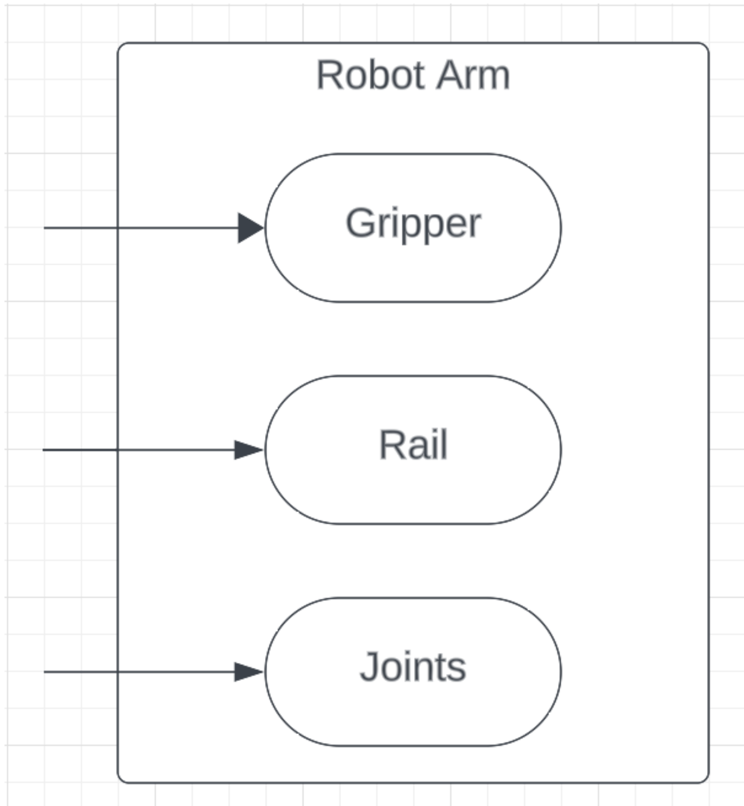

Figure 9: Robot Arm diagram

6.1 Gripper

This layer consists of the gripper which primarily requires the use of the hydraulic suction to create or remove grip to the boxes.

6.1.1 Assumptions

Any wiring required for the gripper is already provided and will be stable to perform the necessary actions for the gripper to execute its task.

6.1.2 Responsibilities

The gripper is responsible for creating grip on to the boxes and remove any grip on to the boxes. The data to create/remove the grip comes from PLC which uses QR code to send the data to hold a grip on to the boxes and once the task is executed, the gripper is given the data to remove the hold on the boxes.

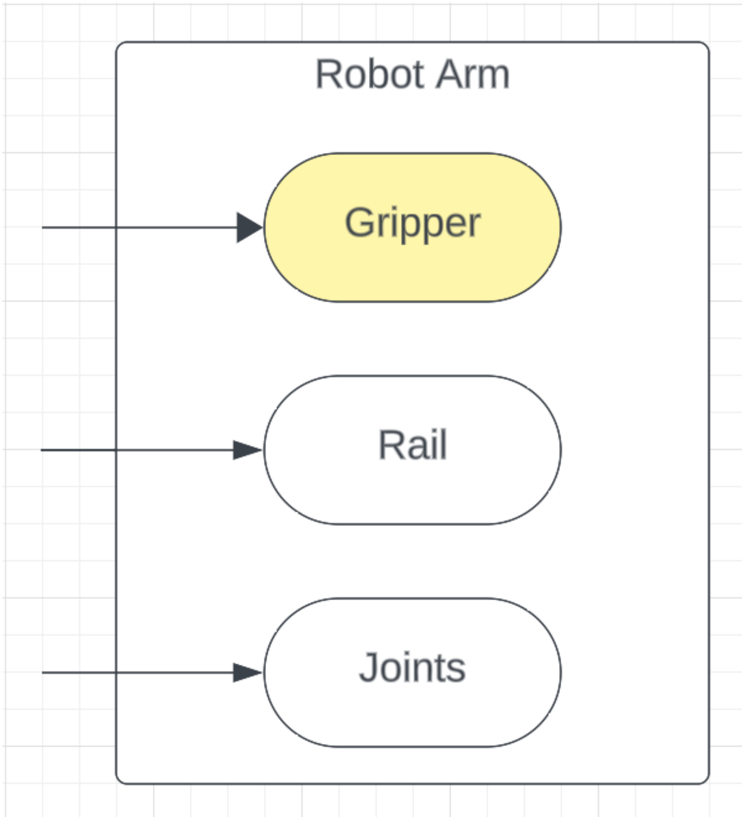

Figure 10: Robot Arm diagram

6.1.3 Subsystem Interfaces

Table 8: Subsystem interfaces

ID	Description	Inputs	Outputs
#1	Suction grip	Box detection	hydraulic grip

6.2 Rail

This layer consists of the linear rail required to give the robot arm a linear motion.

6.2.1 Assumptions

Any wiring required for the motion of linear rail is already provided and is stable when the robot arm is in motion.

6.2.2 Responsibilities

The linear rail is responsible for moving the entire robot arm in a linear motion in defined boundaries in order to get to a configuration to move the boxes. The linear rail will get the data from the servo amplifier which is a component of the PLC.

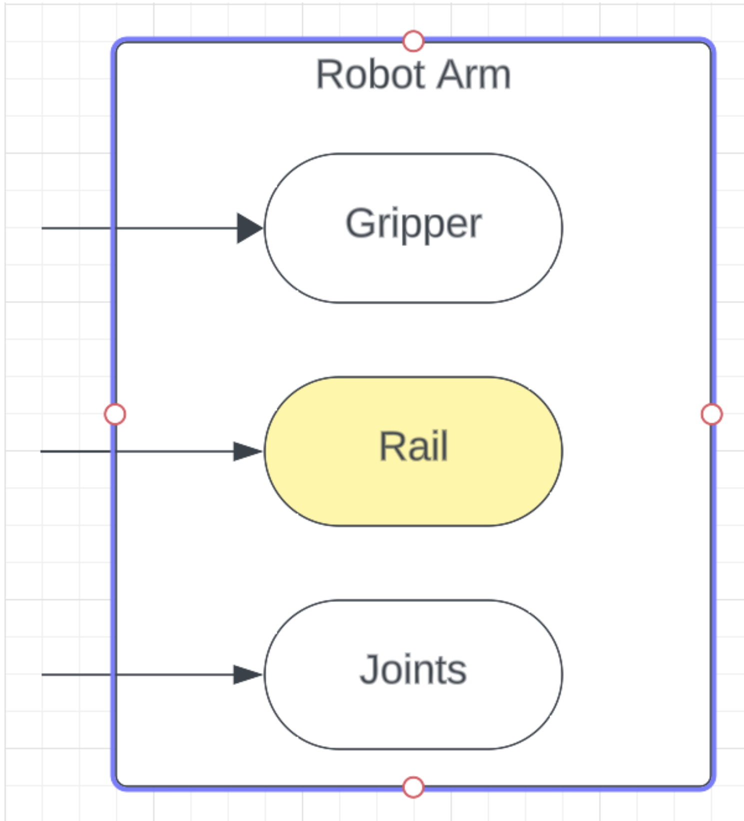

Figure 11: Robot Arm diagram

6.2.3 Subsystem Interfaces

Table 9: Subsystem interfaces

ID	Description	Inputs	Outputs
#2	Rail Motion	PLC data	linear movement

6.3 Joints

This layer consists of the configurations required for the arm to be able to reach the boxes.

6.3.1 Assumptions

Any wiring required for the arm to be in multiple configurations is provided and would be stable while the arm is in motion.

6.3.2 Responsibilities

The joints are responsible for the movement of the gripper and be in a certain configuration so that the gripper is able to reach the boxes. Joints communicate directly with the PLC which gives the appropriate data to be in a certain configuration space.

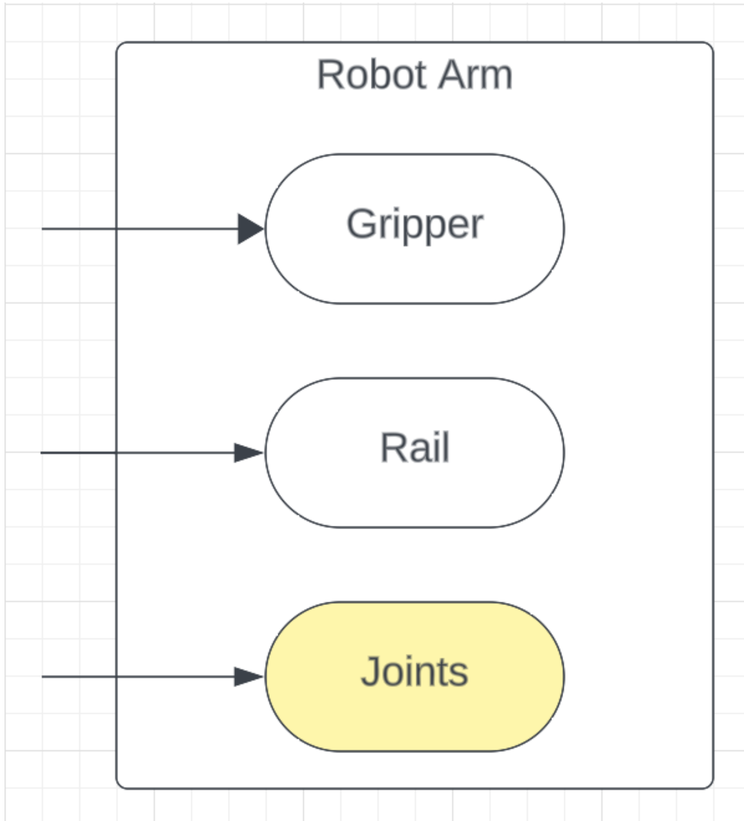

Figure 12: Robot Arm diagram

6.3.3 Subsystem Interfaces

Table 10: Subsystem interfaces

ID	Description	Inputs	Outputs
#2	Joint Configuration	PLC data	joint angle

7 Indicator Layer Subsystems

The indicator layer will be responsible for the output of the system, where the indicator subsystem consists of an industrial light tower, which displays the status of the working cell.

7.1 Industrial Light Tower

An industrial light tower system, typically used in a working cell or industrial setting, serves several essential responsibilities to ensure the efficient and safe operation of the workspace. The specific responsibilities of an industrial light tower system may vary depending on the application. For our system, it serves as an inference to the work environment as it indicates different modes with colored LEDs as get input from the controller and outputs as LEDs.

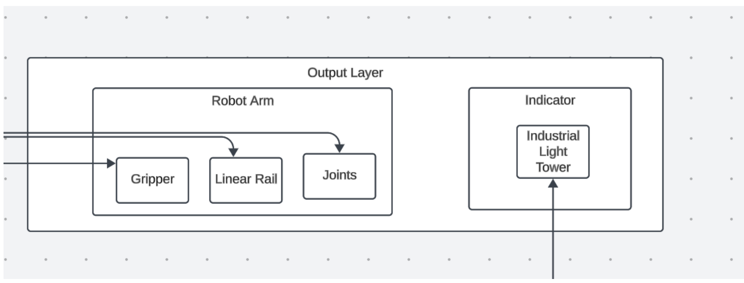

Figure 13: Indicator subsystem description diagram

7.1.1 Assumptions

The "Indicator" subsystem assumes that it will receive status information from the Controller about the working status of the robot arm. It is assumed that this system will be a reliable communication source or interface to update the status of the working cell.

7.1.2 Responsibilities

Signaling and Alerts: Industrial light tower systems often include signal lights or beacons. These lights can be used to convey information or alerts to workers or nearby personnel. Common signals include indicating when a machine is running, signaling the need for maintenance or repair, or warning of specific hazards or emergencies.

Status Indication: The light tower system can provide status indications for machines or equipment within the working cell.

Safety: Ensuring the safety of workers is a primary responsibility. Light towers can be programmed to flash or change colors to draw attention to potential safety hazards, such as moving equipment or areas that require caution.

7.1.3 Subsystem Interfaces

Table 11: Subsystem interfaces

ID	Description	Inputs	Outputs
#01	Stop operating	controller	red led signal
#02	Require caution	controller	yellow led signal
#03	Operating Mode	controller	green led signal

References