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

Detailed Design Specification
CSE 4317: Senior Design II
Spring 2024

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	02.12.2024	AM, MA, AP, HK, KM	Document creation
0.2	05.05.2024	AM, MA, AP, HK, KM	Updating work cell task to painting

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 applying paint to automotive and aerial vehicle parts for various purposes, such as protection, aesthetics and branding. The paint is sprayed using an airbrush. The air brush is connected to an air compressor which is controlled by the programmable logic controller (PLC). This robot arm is set on a linear rail which serves as the 7th joint (additional axis). There are two inductive switches placed on the ends of of linear rail signalling the end of the rail which are also connected to PLC.

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 fixed tasks based on the program. 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.

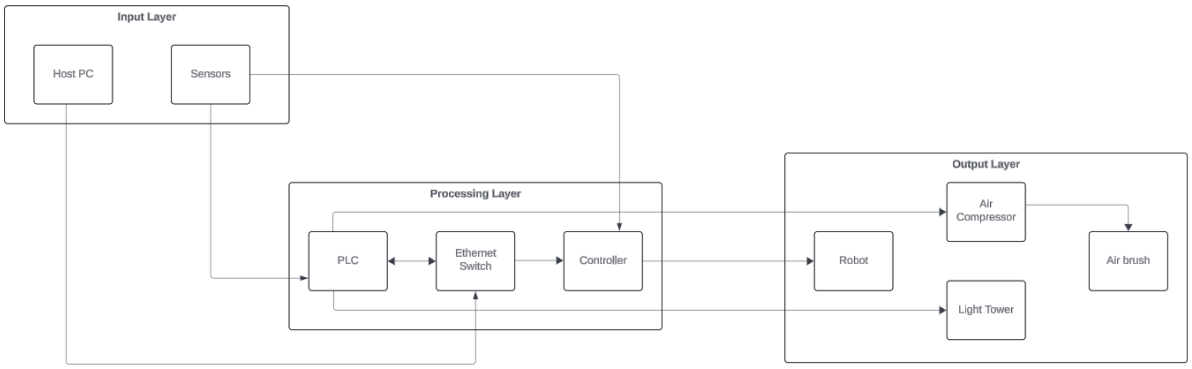

Figure 1: System architecture

3 Input Layer Subsystems

The input layer plays an important part in executing the spray painting task as per the programmed movement. The input in this system comes from the host PC and the sensors. The host PC acts as a centralized location to write programs for robotic movement, as well as control PLC logic. The sensors involved in the system are emergency stops (e-stops) and inductive switches. These inputs send signals to the PLC on particular situations and scenarios to keep the safety measure intact. Inductive switches make sure that the robot arm is calibrated properly before it executes its task. Where as emergency stops immediately halt the robot's operation, ensuring the safety of personnel and preventing potential accidents.

3.1 Layer Hardware

The sensors, including e-stops and inductive switches are the layer hardware for the input layer.

3.2 Layer Operating System

Operating system used for this is Windows 10.

3.3 Layer Software Dependencies

RT Toolbox is required.

3.4 E-Stops

There will be multiple emergency stop (E-stop) buttons strategically placed throughout the system. These buttons will be located next to the case gate, on the controller, and inside the cage. When any of these e-stop buttons are pressed, the entire system will immediately halt, ensuring rapid response to any safety concerns or emergencies.

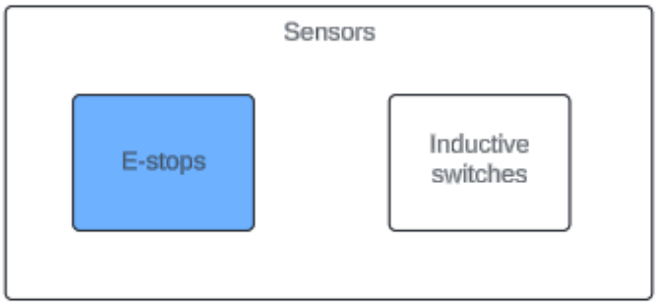

Figure 2: Sensor subsystem diagram

3.4.1 Subsystem Hardware

2 wired emergency stops connected to the CNUSR11 connector of the controller. 1 emergency stop located on the controller.

3.4.2 Subsystem Operating System

N/A

3.4.3 Subsystem Software Dependencies

N/A

3.4.4 Subsystem Programming Languages

N/A

3.4.5 Subsystem Data Structures

N/A

3.4.6 Subsystem Data Processing

The signals are send through wires to robot controller for processing to halt the robot.

3.5 Inductive Switches

Inductive switches are placed on the edges of linear rail to mark the endpoints. Using these switches will help the robot to calibrate the center of linear rail whenever robot starts.

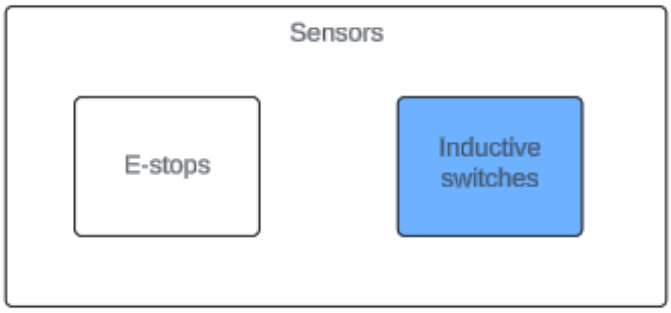

Figure 3: Sensors subsystem diagram

3.5.1 Subsystem Hardware

The inductive proximity sensor is a barrel-type sensor with PNP output, featuring a 1.5mm sensing range. It operates with a 3-wire configuration and is rated IP67. These sensors are typically connected to the general-purpose input/output pins of the PLC.

3.5.2 Subsystem Operating System

Windows 10.

3.5.3 Subsystem Software Dependencies

GX Works is needed in order to write ladder logic that processes the signals to act appropriately.

3.5.4 Subsystem Programming Languages

MELFA-BASIC VI programming language.

3.5.5 Subsystem Data Structures

N/A

3.5.6 Subsystem Data Processing

The switch sends a signal to the PLC when the robot passes over it.

3.6 RT Toolbox

The RT Toolbox is a software package developed by Mitsubishi Electric for their industrial robots. It serves as a comprehensive toolset to assist with various tasks related to robot programming, configuration, and maintenance

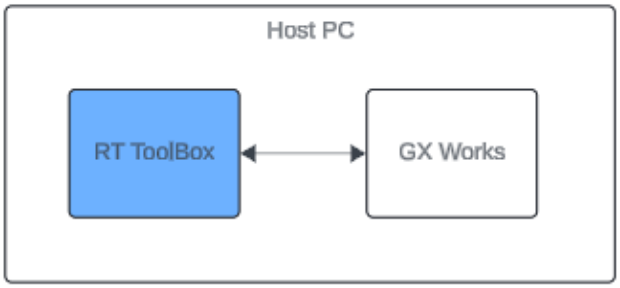

Figure 4: Host PC subsystem diagram

3.6.1 Subsystem Hardware

Host PC acts as system hardware in which RT Toolbox operates.

3.6.2 Subsystem Operating System

Windows 10.

3.6.3 Subsystem Software Dependencies

N/A

3.6.4 Subsystem Programming Languages

MELFA BASIC VI.

3.6.5 Subsystem Data Structures

Synchronous execution of the code.

3.6.6 Subsystem Data Processing

Utilizing Host PC's memory.

3.7 GX Works

GX Works3 is the configuration software for FX, L, and Q Series controllers developed by Mitsubishi Electric. GX Works3 interfaces with the work cell's programmable logic controller to program in several languages like ladder logic.

3.7.1 Subsystem Hardware

Host PC acts as system hardware.

3.7.2 Subsystem Operating System

Windows.

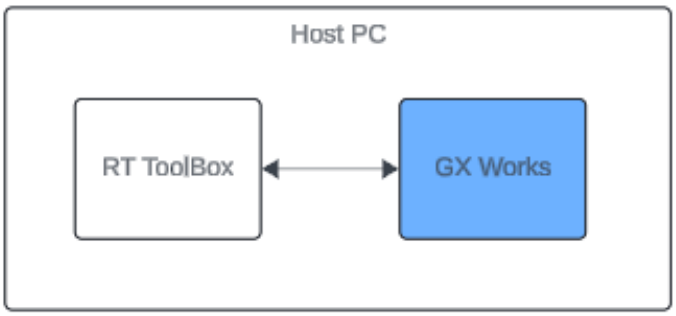

Figure 5: Host PC subsystem diagram

3.7.3 Subsystem Software Dependencies

N/A

3.7.4 Subsystem Programming Languages

Ladder Diagram (LD): A graphical representation of relay logic.
Structured Text (ST): A high-level textual language based on Pascal.
Function Block Diagram (FBD): A graphical language using function blocks.
Instruction List (IL): A low-level textual language.
Sequential Function Chart (SFC): A graphical language for sequential control.

3.7.5 Subsystem Data Structures

N/A

3.7.6 Subsystem Data Processing

Using Host PC's memory.

4 Processing Layer Subsystems

In this section, the processing layer's hardware and software design are delineated. The layer comprises a PLC, Ethernet switch, and a Robot Controller interconnected to facilitate communication and decision-making processes. The processing layer comprises of connection and communication between the PLC and the robot controller primarily. In order to program the PLC to do the required tasks, GX Works3 (software) and RTToolbox (software) are jointly used. A program written in ladder logic controls the logic for the air compressor and light tower. The program inside the PLC is written from the PC and executed. The instructions are passed onto the controller and directs the robot to move its joints according to the program.

4.1 Layer Hardware

The hardware involved with the processing layer include a MELSEC Programmable Logic Controller, issued by Mitsubishi. The PLC is a specialized computer used in industrial automation and is paired with software to make decisions based on the instruction from the Host PC, signals from the switches. Finally, the PLC is equipped with an ethernet socket that enables communication between the Host PC and PLC through an Ethernet switch. The robot controller controls the actions and behaviors of robotic systems through joints movement, transforming programmed instructions into precise and efficient movements to fulfill a wide range of industrial tasks and applications.

4.2 Layer Operating System

The operating systems required by the layer are dependent on the specific components. The host PC runs on Windows 10 and PLC is on Real Time Operating System (RTOS).

4.3 Layer Software Dependencies

The Host PC uses RT ToolBox for ladder logic, GX Works3 to controlled movement of joints, and it uses Windows 10 as operating system.

4.4 PLC

PLCs provide a centralized platform for controlling and programming the movements of the RV-8 robot. They execute logic sequences that regulate the robot's motions, trajectories, and speeds, ensuring precise and coordinated operation in the manufacturing environment. PLCs interface with sensor systems to enable adaptive control schemes and provide input to RV-8 robots. They have safety features such emergency stop connections to validate the work cell's safety state and light towers to show the operating condition.

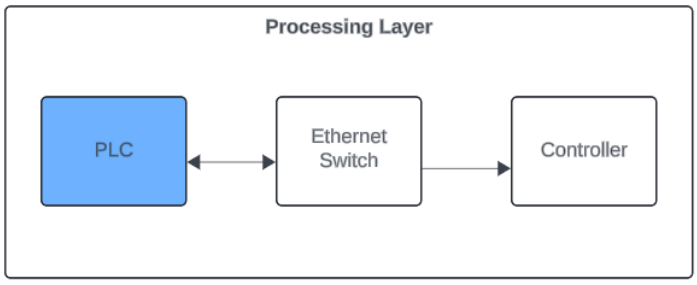

Figure 6: PLC Subsystem

4.4.1 Subsystem Hardware

The Mitsubishi Electric MELSEC iQ-F FX5UC-32MT/DSS-TS PLC is a compact unit featuring 32 I/O points, 16 digital inputs, and 16 digital outputs, all operating at 24Vdc and Ethernet communication capabilities. The PLC also has an external relay switch FX5-8EYR/ES with 8 digital outputs.

4.4.2 Subsystem Operating System

Operating on a real-time operating system (RTOS), the PLC ensures precise timing and reliable execution of control logic.

4.4.3 Subsystem Software Dependencies

Software tools such as GX Works and MRConfigurator2, provided by Mitsubishi Electric, enable the programming and configuration of the PLC for control tasks.

4.4.4 Subsystem Data Processing

The PLC processes sensor feedback and command signals in real-time, executing control logic to coordinate the movements of the robotic arm and additional axis. Communication with the host PC via TCP/IP facilitates efficient data exchange for seamless system operation.

4.5 Ethernet Switch

Ethernet Switch plays an important role in facilitating communication and data exchange within the control system. In order to provide compatibility with a variety of devices and controllers and to enable smooth integration with current automation systems, the switch supports a wide range of communication protocols, including TCP/IP.

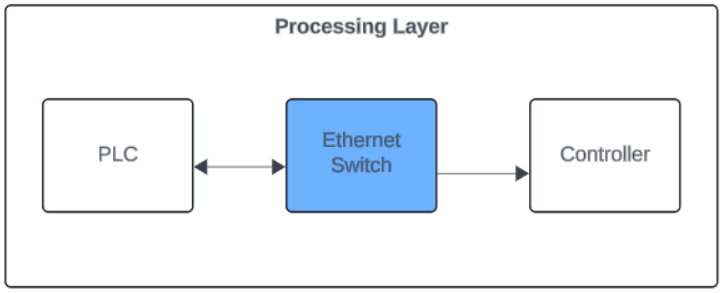

Figure 7: Ethernet Switch Subsystem

4.5.1 Subsystem Hardware

The subsystem uses Ethernet switch board with 8 sockets.

4.5.2 Subsystem Software Dependencies

Uses Windows for it's user-interface.

4.6 Robot Controller

The movements, actions, and functions of one or more robotic arms or manipulators are coordinated and controlled by a robot controller, which acts as the control system of a robotic system. Controlling the robotic arm's motion is the main responsibility of the robot controller. The robot's joints are controlled by motors and actuators that it creates commands to drive, allowing for precise and coordinated movement

in three dimensions. It support TCP/IP communication protocol for interfacing with other devices and robot arm to communicate. It is connect with PLC via Ethernet switch, which acts as slave device and PLC as master device, as PLC processes the data from the slave device to performs control actions using robot controller. The back panel of the robot controller has 2 connectors (CNUSR11 and CNUSR12) pins that allow for other connections, such as emergency stop (e-stop) capabilities, which guarantees rapid stops to the robot arm's movements in an emergency. It also makes it easier to attach a light tower, which provides a visible signal of the operational state of the working cell. The robotic system's overall safety and efficiency are improved by this combination of safety features, which offer real-time monitoring and quick response times to guarantee a safe working environment.

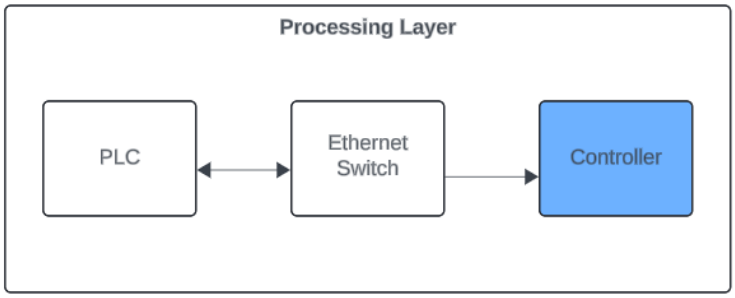

Figure 8: Robot Controller Subsystem

4.6.1 Subsystem Hardware

It contains network card for CC-Link.

4.6.2 Subsystem Operating System

Operates on Host PC, which uses Windows as it's operating system.

4.6.3 Subsystem Software Dependencies

It uses RT-ToolBox for it's programming, also called Ladder logic programming.

4.6.4 Subsystem Programming Languages

Programming languages supported by GXWorks, including ladder logic, function block diagrams, and structured text, are utilized for developing control algorithms tailored to the application. RT ToolBox is written in MELFA BASIC VI.

4.6.5 Subsystem Data Processing

No data processing is done by Robot controller, it uses operates on PLC logical output.

5 Output Layer Subsystem

This section consists of all the components needed to produce desired output. The major components are the robot, air compressor, air brush and industrial light tower. The robot receives instructions from controller in order to move the joints and linear rail. Whereas, PLC triggers air compressor and light tower when it is required.

5.1 Layer Hardware

The robot, air compressor, airbrush, and light tower collectively constitute the hardware layer

5.2 Layer Operating System

Windows.

5.3 Layer Software Dependencies

GX Works and RT Toolbox.

5.4 Robot Arm

The robot arm consists of six distinct joints that move in specific directions, enabling the robot to perform the desired tasks. It includes capabilities for 2 additional axes.

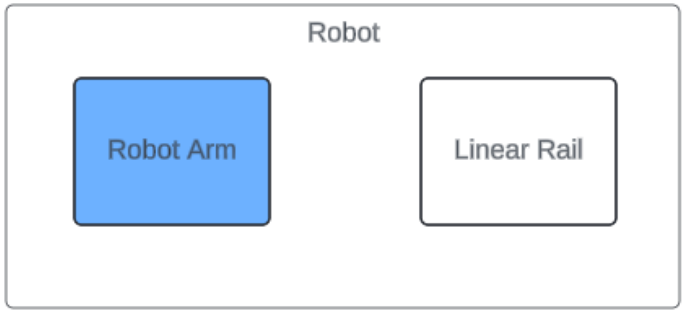

Figure 9: Robot Subsystem

5.4.1 Subsystem Hardware

The robot arm is a Mitsubishi RV-8CRL robot with an 8 kilogram payload.

5.4.2 Subsystem Operating System

N/A

5.4.3 Subsystem Software Dependencies

RT Toolbox.

5.4.4 Subsystem Programming Languages

MELFA BASIC VI.

5.4.5 Subsystem Data Structures

N/A.

5.4.6 Subsystem Data Processing

Signals from controller to robot arm to perform the movement, as well as signals from the PLC.

5.5 Linear Rail

Linear rail is being used as an additional axis to provide the robot a movement along an axis. It is considered to be the 7th axis of the robot in this case.

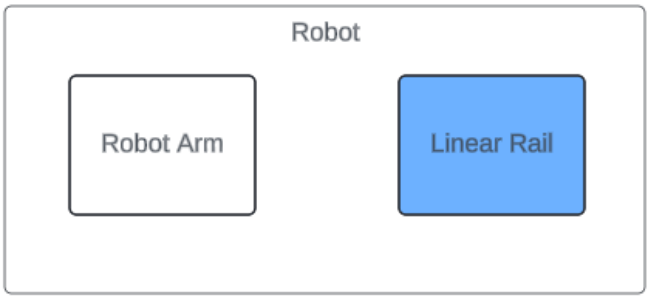

Figure 10: Robot Subsystem

5.5.1 Subsystem Hardware

Linear rail itself.

5.5.2 Subsystem Operating System

N/A.

5.5.3 Subsystem Software Dependencies

RT Tool box.

5.5.4 Subsystem Programming Languages

MELFA BASIC VI.

5.5.5 Subsystem Data Structures

N/A.

5.5.6 Subsystem Data Processing

Signals from controller to robot arm to perform the movement.

5.6 Air Compressor

An air compressor is a pneumatic device that converts power (using an electric motor, diesel, or gasoline engine) into potential energy stored in pressurized air (also known as compressed air).

5.6.1 Subsystem Hardware

Air Compressor, as well as a solenoid switch triggered by PLC relay. Upon contact, air can flow through the pneumatic line to the robot and air brush.

5.6.2 Subsystem Operating System

N/A.

5.6.3 Subsystem Software Dependencies

GX Works.

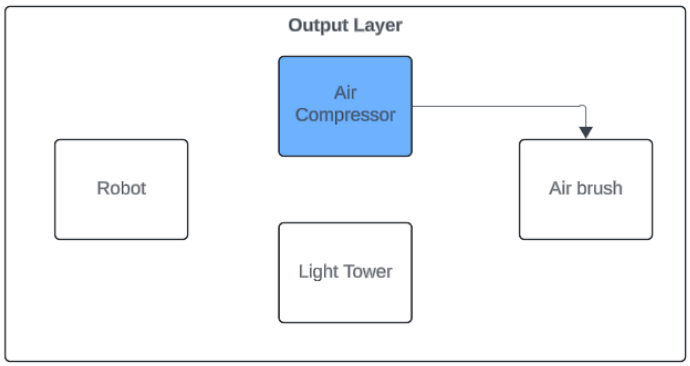

Figure 11: Air Compressor Subsystem

5.6.4 Subsystem Programming Languages

Ladder diagram/logic.

5.6.5 Subsystem Data Structures

N/A.

5.6.6 Subsystem Data Processing

Signals from PLC to Air Compressor when to release pressure.

5.7 Light Tower

A light tower is a piece of mobile equipment that combines high-intensity electric lamps with a mast.

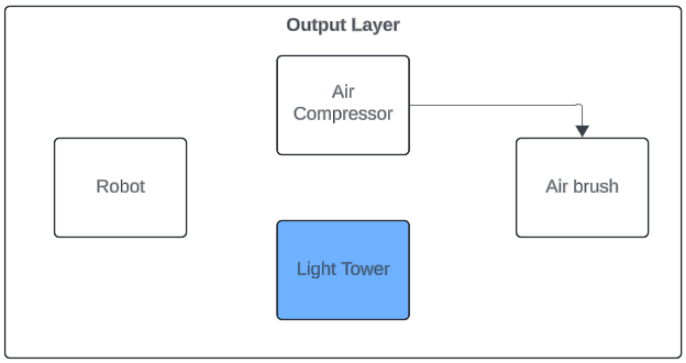

Figure 12: Pneumatic Control Subsystem

5.7.1 Subsystem Hardware

Light tower.

5.7.2 Subsystem Operating System

**Windows.

5.7.3 Subsystem Software Dependencies

GX Works.

5.7.4 Subsystem Programming Languages

Ladder Design/Logic.

5.7.5 Subsystem Data Structures

N/A.

5.7.6 Subsystem Data Processing

Signals flow from PLC to light tower indicating the status of robot.

5.8 Air Brush

An airbrush is a small, air-operated tool that atomizes and sprays various media, most often paint, but also ink, dye, and foundation.

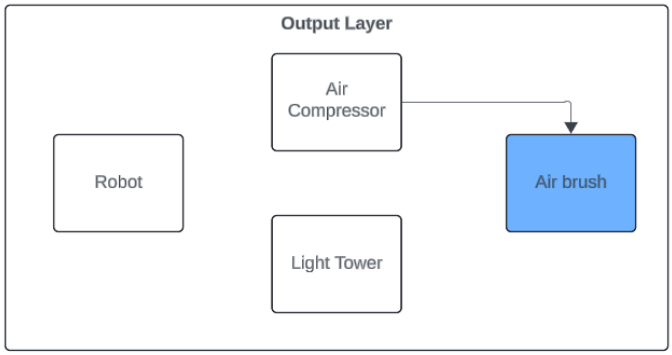

Figure 13: Pneumatic Control Subsystem

5.8.1 Subsystem Hardware

**Air brush.

5.8.2 Subsystem Operating System

N/A.

5.8.3 Subsystem Software Dependencies

N/A.

5.8.4 Subsystem Programming Languages

N/A.

5.8.5 Subsystem Data Structures

N/A.

5.8.6 Subsystem Data Processing

There is no data which is being processed. Although air compressor sends air pressure when it is triggered by the PLC. The air pressure is passed through pneumatic lines.

6 Appendix A

References