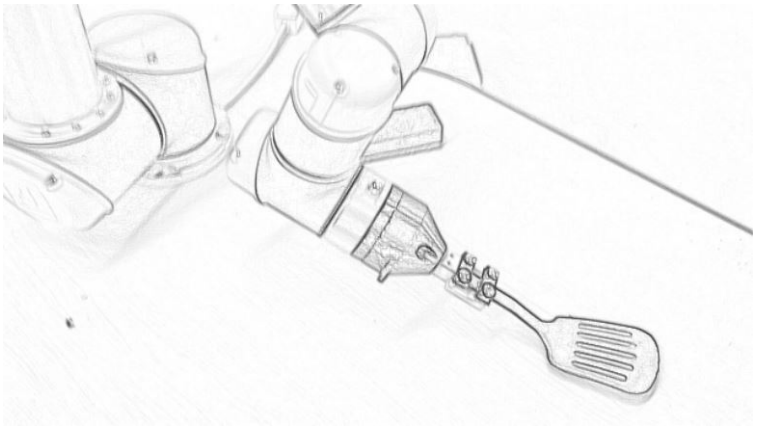
Department of Computer Science & Engineering
The University of Texas at Arlington

Architectural Design Specification
CSE 4317: Senior Design II
Spring 2018



Team Tassium
Master Chef

Anthony Tatowicz
Jesse Daniel Mitchell
Todd Brewer
Linh Vu

Revision History

Revision	Date	Author(s)	Description
0.1	12.04.2017	LV, JDM	First draft
0.2	5.09.2018	LV, TB	Revise documents, add pictures
0.3	5.11.2018	TB	Final Draft

CONTENTS

List of Figures

List of Tables

1 INTRODUCTION

The UR5 Master Chef is a product that aims to alleviate the need for manual labor in a cooking environment. The purpose of this design is to provide an interchangeable tool interface allowing for the use of various tools for use in a fast paced food preparation area. The target application for his prototype is to demonstrate the creation, preparation, and serving of grill cheese sandwiches.

2 System Overview

The main architecture of the system involves 4 main systems communicating with each other. The control logic system and UR5 system pass information with each other via the network system through an Ethernet TCP/IP connection. The UR5 then provides a signal to the mount system allowing for the docking and un-docking of various utensils.

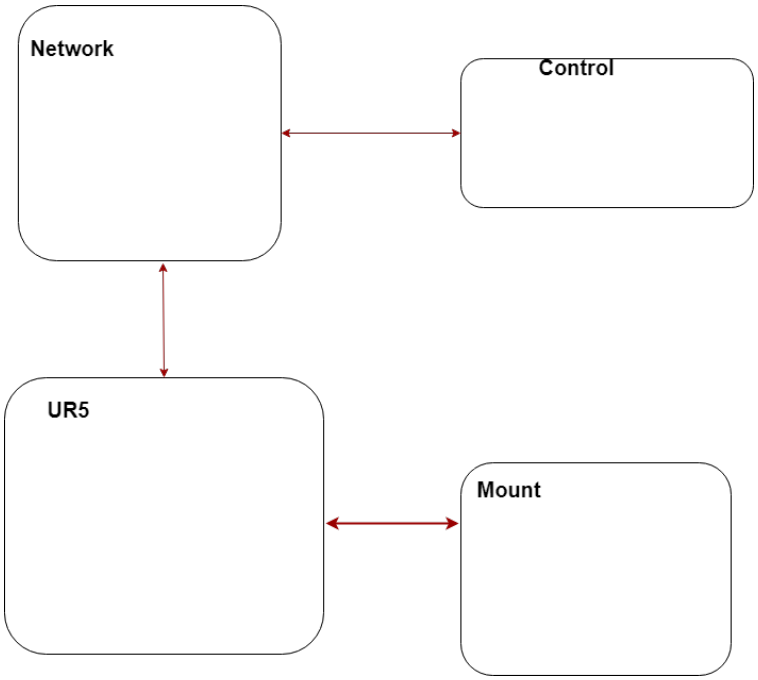


Figure 1: A simple architectural layer diagram

2.1 Network

This layer contains the router that will allow a connection to the UR5.

2.2 Control

This layer contains the camera, raspberry pi that will be use to communicate with the UR5.

2.3 Mount

This layer contains the mount that will fit the tool into the mount by controlling the magnet.

2.4 UR5

This layer contains the UR5 robot arm, the Polyscope (interface), and the control box.

3 Subsystem Definitions & Data Flow

This section breaks down your layer abstraction to another level of detail. Here you graphically represent the logical subsystems that compose each layer and show the interactions/interfaces between those subsystems. A subsystem can be thought of as a programming unit that implements one of the major functions of the layer. It, therefore, has data elements that serve as source/sinks for other subsystems. The logical data elements that flow between subsystems need to be explicitly defined at this point, beginning with a data flow-like diagram based on the block diagram.

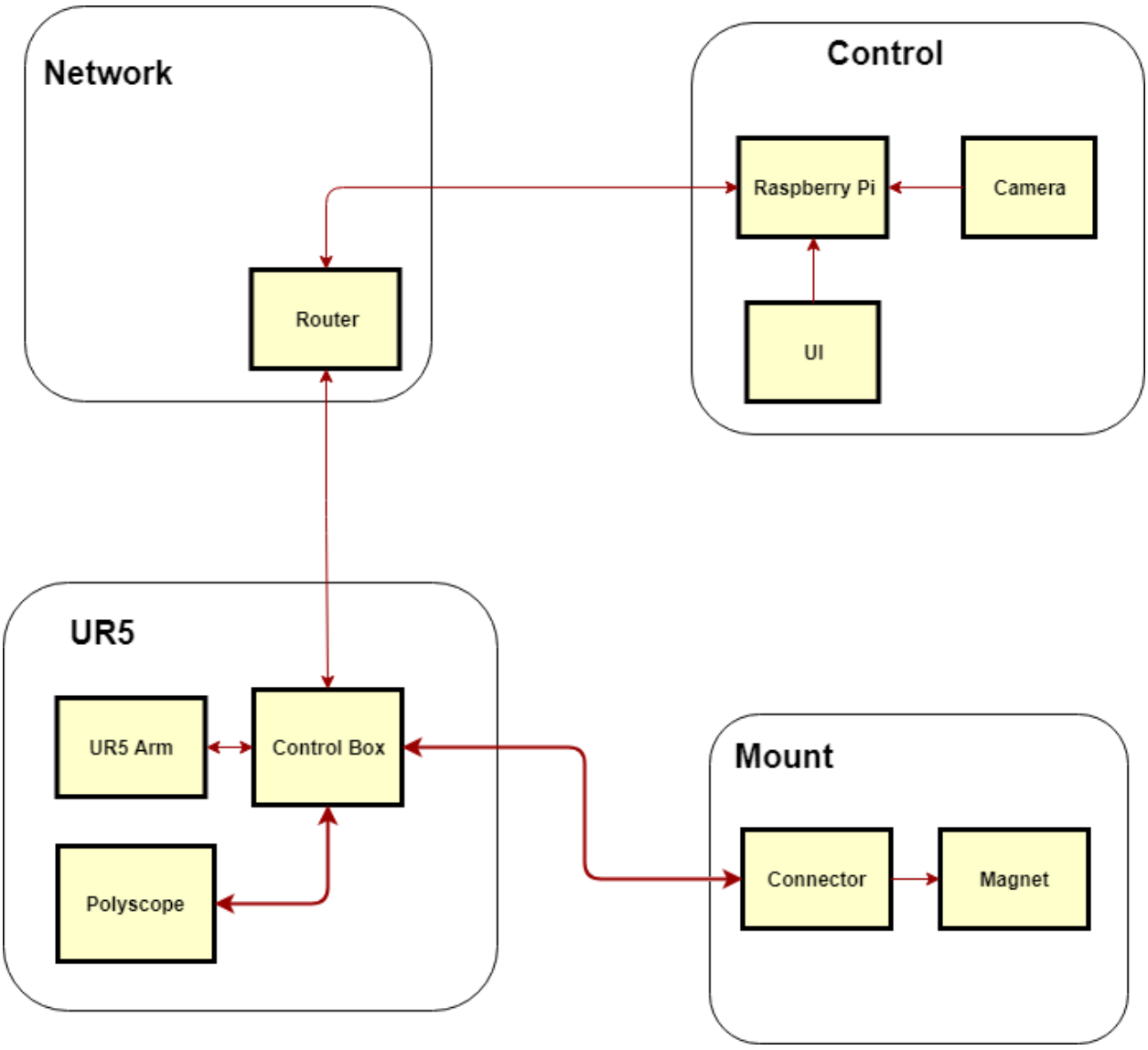


Figure 2: A simple data flow diagram

4 Network Layer Subsystems

This layer is for communication between the control layer and the UR5 layer in details. It have a router as its subsystem.

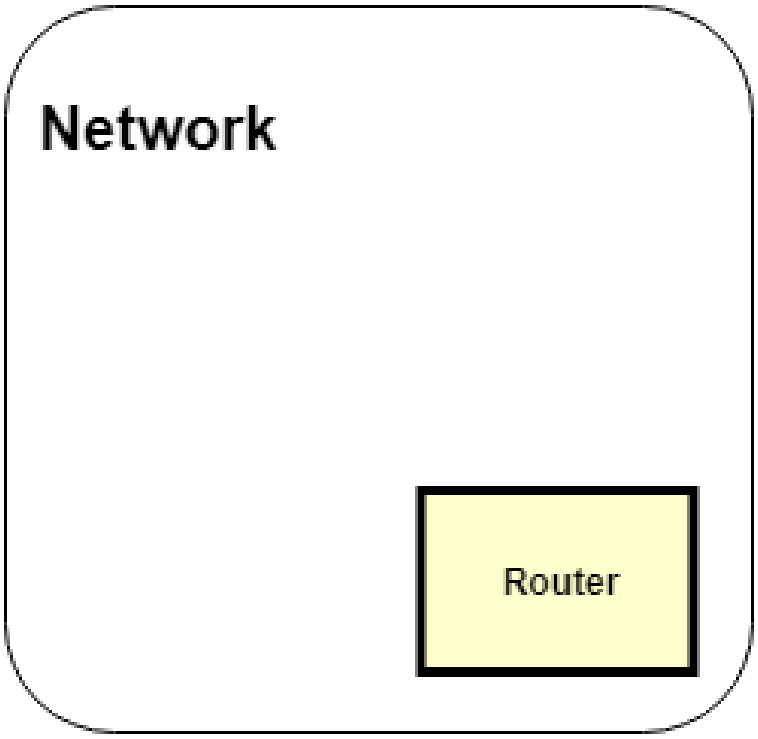


Figure 3: Network layer diagram

4.1 Router

The router is use as a mean of communication to the UR5.

4.1.1 Assumptions

The IP address is given and known to the UR5 through the Polyscope.

4.1.2 Responsibilities

The router gives an IP address that allow the software to send instructions to the UR5.

4.1.3 Subsystem Interfaces

There is two interfaces connecting to the router.

Table 2: Router interfaces (through Ethernet connection

ID	Description	Inputs	Outputs
#1	Control Box	N/A	UR Script (Robot instructions via TCP/IP)
#2	Raspberry Pi	UR Script (Robot instructions via TCP/IP)	N/A

5 Control Layer Subsystems

This layer is the integration of the UI and camera in details.

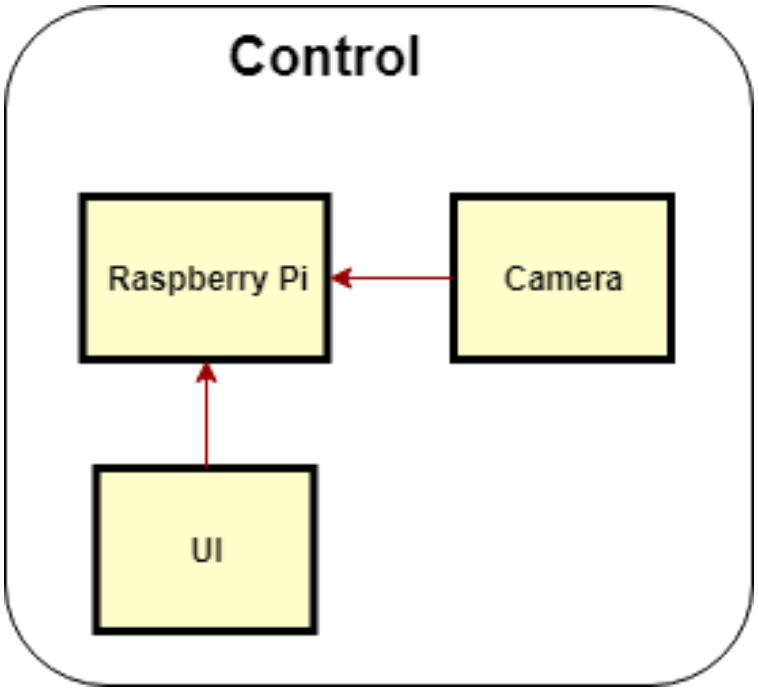


Figure 4: Control layer diagram

5.1 User Interface (UI)

The user interface operates through a browser that allow the user to give specifics task to the UR5. User interaction is via a monitor and mouse attached to the Raspberry Pi.

5.1.1 Assumptions

The browser the user will be using is Google Chrome. Other browser may work just as well. This project, however, tested and developed the UI through Google Chrome.

5.1.2 Responsibilities

The purpose of this UI is to allow the user to gives specific command to the UR5. The commands are as follows:
Home : Move UR5 to default position.
Start : Make grill cheese sandwich.
Stop : Cease function.
Rack Tool : Put away tool.
Grab Tool : Grab tool.

*The tool for this project specifically is a spatula. Other tools will be implement later in the future.

5.1.3 Subsystem Interfaces

The UI connects to the Raspberry Pi.

Table 3: UI interfaces

ID	Description	Inputs	Outputs
#1	Raspberry Pi	N/A	UI Commands (via mouse)

5.2 CAMERA

The camera use to determine the location of certain object(sandwich) or if there is another object in the way of the UR5.

5.2.1 ASSUMPTIONS

The camera is a 720p or higher. At least one camera.

5.2.2 RESPONSIBILITIES

The purpose of the camera is to determine the location of the sandwich relative to its position so that the UR5 can pick up the sandwich. Its other responsibility is also to detect foreign obstacle or human being so that it can stop before colliding.

5.2.3 SUBSYSTEM INTERFACES

The camera connects to the Raspberry Pi.

Table 4: Camera interfaces

ID	Description	Inputs	Outputs
#1	Raspberry Pi	N/A	Location of sandwich Detect foreign obstacle or human being

5.3 RASPBERRY PI

The Raspberry Pi houses all of the programs produced in this project. It generates the UI via a node.js server for the user to interact with and output robot commands to the network system based on user input.

5.3.1 ASSUMPTIONS

That all hardware connections are supported and configured properly: mouse and monitor are set up, and network connection to the UR5 is established.

5.3.2 RESPONSIBILITIES

To handle all inputs from the user and outputs to the network.

5.3.3 SUBSYSTEM INTERFACES

Table 5: Raspberry Pi interfaces

ID	Description	Inputs	Outputs
#1	UI	UI Commands	N/A
#2	Camera	Location of sandwich Detect foreign obstacle or human being	N/A

6 UR5 Layer Subsystems

This layer is the destination of all the instructions from the control layer go to. It contains the UR5's arm, its control box, and a Polyscope for the UI.

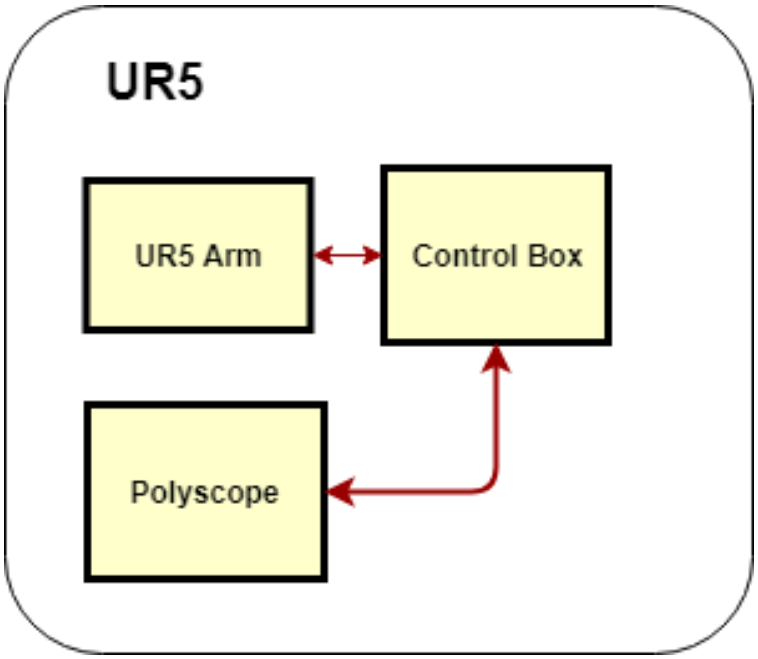


Figure 5: UR5 layer diagram

6.1 UR5 Arm

This is the physical arm of the UR5. It receives instruction from the control box and act accordingly.

6.1.1 Assumptions

The arm is fully functional and can pick up at least 5 pounds. It is connected to the control box. It comes with the control box and Polyscope.

6.1.2 Responsibilities

This arm takes in instruction from the UI. It moves accordingly to what the user instructions through the UI.

6.1.3 Subsystem Interfaces

The arm is connected to the control box as its interface.

Table 6: UR5 Arm interfaces

ID	Description	Inputs	Outputs
#1	UR5 control box	Instructions(UR script)	Position of arm

6.2 Control Box

The control box is what allow the arm to receive instructions and move accordingly. It receives instruction from the UI through the Polyscope or it can receive instructions from an outside source through a socket.

6.2.1 Assumptions

Functional and connects to the arm. It comes with the arm and Polyscope.

6.2.2 Responsibilities

The control box receive inputs and then outputs the instruction to the arm.

6.2.3 Subsystem Interfaces

The control box is connected to the arm and the Polyscope. It is also controlling the connector that is on the wrist of the arm.

Table 7: Control Box interfaces

ID	Description	Inputs	Outputs
#1	UR5 Arm	Position of Arm	Waypoints UR scripts
#2	Polyscope	Instructions(UR script) Waypoints I/O control Voltage	Position of arm I/O control Voltage
#3	Connector	N/A	Voltage

6.3 Polyscope

The Polyscope is a UI that comes with the UR5. It is where all the script and control of the UR5 movements that can be instructed by writing a script.

6.3.1 Assumptions

The user is not using the Polyscope as a way of sending the the instructions to the arm. It is connected to the control box. It comes with the control box and the arm.

6.3.2 Responsibilities

The emergency stop button is use to stop the arm immediately.

6.3.3 Subsystem Interfaces

The Polyscope interface is the control box.

Table 8: Polyscope interfaces

ID	Description	Inputs	Outputs
#1	UR5 control box	Position of arm I/O control Voltage	Instructions(UR script) Waypoints I/O control Voltage

7 Mount Layer Subsystems

This layer is the mount that control the release and hold of the tool.

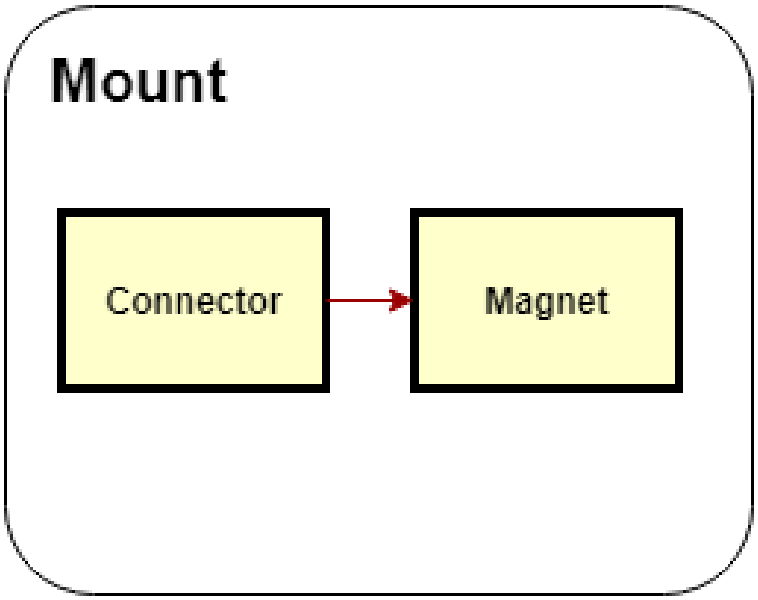


Figure 6: Mount layer diagram

7.1 Connector

The connector is mounted on the wrist of the arm, but it is being control by the control box.

7.1.1 Assumptions

There is a 8 pin connector at the wrist of the arm and an available Lumberg RKMV 8-354 cable for the 8-pin connector.

7.1.2 Responsibilities

This connector control the mount to grab or release the tool by sending the magnet either 12V or 0V respectively.

7.1.3 Subsystem Interfaces

The connector interacts with the magnet by the instructions of the control box.

Table 9: Connector interfaces

ID	Description	Inputs	Outputs
#1	Control Box	Voltage(0V or 12V)	N/A
#2	Magnet	N/A	Voltage(0V or 12V

7.2 Magnet

The magnet is place inside the mount where the tool can be inserted.

7.2.1 Assumptions

The magnet is strong enough to hold at least 5 pounds. It receives either 0V or 12V from the UR5. It does not takes more than 12V.

7.2.2 Responsibilities

The magnet is to grab/hold or release the tool by receiving either 0V or 12V.

7.2.3 Subsystem Interfaces

The magnet interface with the connector.

Table 10: Magnet interfaces

ID	Description	Inputs	Outputs
#1	Connector	Voltage(0V or 12V)	N/A