# Keyboard Interface Design Rev 1.1

## **Name \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

## **Partner(s) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

## **Goal:**

Part 1 is a group activity where you will discuss and define an interface for your piRover. First you review the control interface from the smart phone application and identify all required rover actions. Next, you will discuss the actions and determine how these can be emulated using the keyboard. In other words, how can you map actions that will be eventually entered in the app to keyboard actions? You use the keyboard emulation as you test your final project. You will list all actions in a table and specify keyboard inputs required for each.

In part 2, you will create the Python program that will capture the inputs specified in part 1 and simply print a message indicating the type of motion or action that is being requested. See the selection structure presented in the User Move solution. Using W, X, A, and D is okay, but you can also define others.

## **Prerequisites:**

This assessment requires content and code solutions from the following:

* User Move
* PWM Introduction

## Performance Outcomes:

1. Design a keyboard interface for all piRover actions defined on the Control screen of the smartphone application.
2. Create user input and selection code that implements interface design.

## Resources:

1. See prerequisite lessons

## Materials:

1. piRover

## Set Up

1. Prepare your workspace for this project.
   1. Connect to your piRover using VNC. Access your piRover folder and launch VS Code.
   2. Create a directory for this week’s work if required.
   3. Create a new **piRover\_keyboard.py** file in this week’s directory.

## Part 1 – Keyboard Interface Design

1. Your task is to work with your group to define a keyboard interface for all piRover actions.
2. Identify the required actions by reviewing the Control screen on the smart phone application. An image is provided below. The instructor will discuss the operation of slider inputs. A plan will be determined for how to interact with these analog inputs.

Diagram

Description automatically generated

1. With your partner or team, discuss the use of the input() function and how keystrokes along with the Enter Key can be used to create a drive interface. complete table 1 below. Note that not all identified items in the image above are control inputs. Included only inputs to the piRover in the table below.

Table 1

|  |  |  |
| --- | --- | --- |
| **Action ID** | **Action Name** | **Keyboard Input** |
| A | FORWARD | W |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

1. Be prepared to share your design with the class.

## Part 2 – Keyboard Drive

1. Create the **piRover\_keyboard.py** code that implements the user interface defined in Table 2. This is the “main” solution file that does the following
   1. Welcomes the user to the keyboard move solution
   2. Provides directions to the user including a list of keyboard commands. Create a menu() function that provides this feature.
   3. Captures the command
   4. Use a selection structure to print an appropriate movement action.
   5. Includes a method for the user to exit the solution. See the User Move solution for options.
   6. No implementation of actions is required. This comes later.

## Submission:

1. Submit this document with Table 1 completed providing keyboard inputs for all piRover control actions. Submit the piRover\_keyboard.py file that implements the command input, selection, and required print statements.