AIUB Robotic Crew

# Interface between Hardware & Software

NETWORK ARCHITECTURE DIAGRAM



Here we are using TCP network protocol to communicate with 3 raspberry pi’s and the user control software. Raspberry pi’s are acting as server and the control software is acting as the client. Three raspberry pi’s are serve different purpose. Control software is acting as client.

# TASKS OF DIFFERENT RASPBERRY PI’S

1. RASPBERRY PI 1 (Rover movement)

* ARDUINO 1 (Rover movement control)
* It is controlling the rover motors which are responsible for movement of the rover.
* ARDUINO 2 (Hand movement control)
  + This Arduino is controlling the hand of the rover.
* Controlling the other rover basic operation.

1. RASPBERRY PI 2 (Image processing)
   * This raspberry pi is connected with all the camera module.
   * Science task is done here.
   * Secondary database is host in this raspberry pi.
2. RASPBERRY PI 3 (Backup)
   * This raspberry pi will be Standby.
   * If any of Raspberry pi or Arduino is goes down this raspberry pi will be in operation.
   * This Raspberry pi is responsible for all the automated decision.
   * If the rover is disconnected from the base this raspberry pi will get the rover back in the route.
   * It is also responsible for image processing.

# NETWORK COMMUNICATION

We are using TCP communication protocol for the communication with all the raspberry pi and the user control software. In this communication protocol we are using a port to pass and receive our data. And to communicate with Arduino and raspberry pi we are using Firmata protocol.

# Rover Control Software

Rover will be controlled from base by the control software. It is a combined package of many different module which are responsible for different activities of the rover. Each module is independent in their tasks. Like rover movement module is responsible for movement of rover. Modules can internally communicate with other modules to perform their individual tasks. Also there is a console module which can show all the commands currently processed by control software.

## Modules:

### Rover Movement Control Module

### Rover Visualization Module

### Arm Visualization Module

### Web Camera View Module

### Connection Settings & Maintenance Dock

### Rover Feedback Module

### Rover Sensor Data Module

### Rover Console Module

### GPS Navigation Module

### Connector One Module

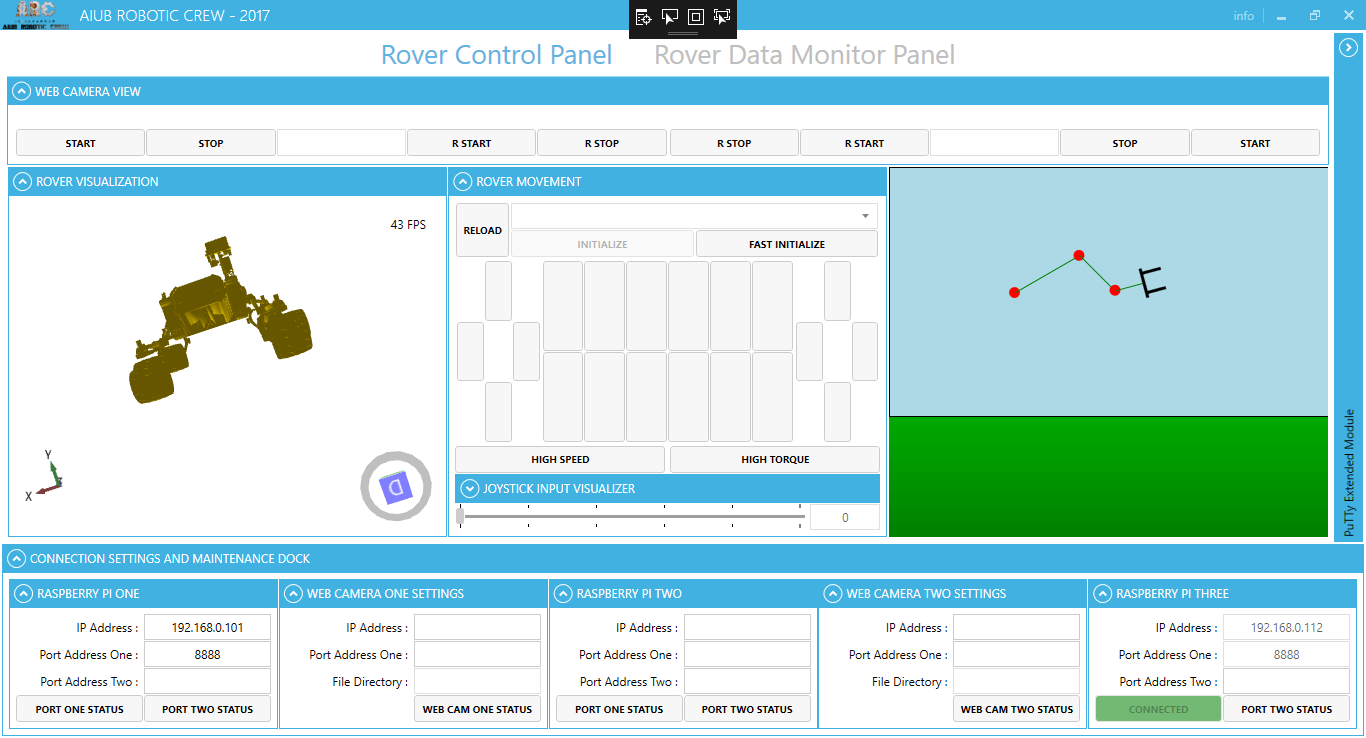


Figure 1: Rover Control Software

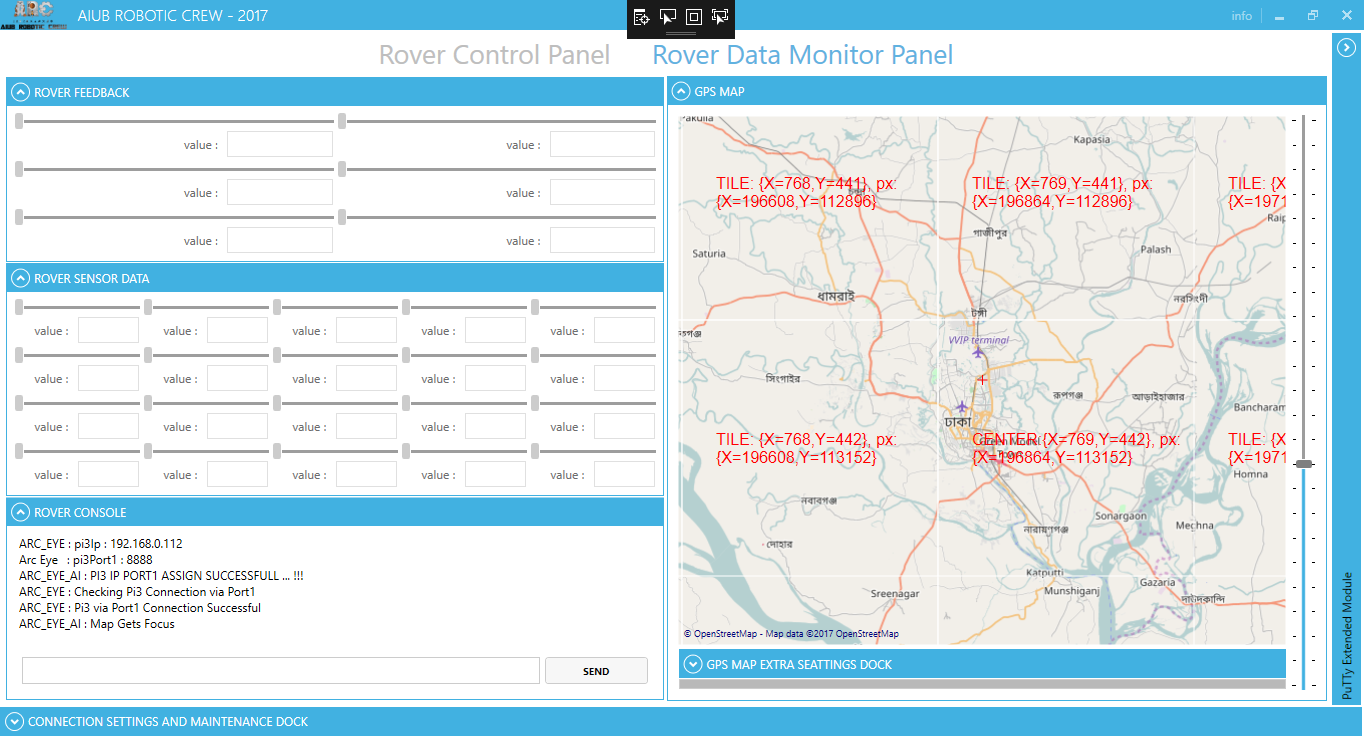


Figure 2: Rover Control Software

# Control Software Architecture

All the module are fully independent from each other they are running in different thread so that there always the main thread is free to take command from the operator. And collect data from different modules. Architecture of the control software.



Figure 3: Control Software Architecture

# Rover Movement Control Module

Rover Control Module is responsible for each and every movement of the rover. This module can receive command from Keyboard & Mouse or Joystick. If anything goes wrong with joystick by default it will convert in Keyboard & Mouse mode. This module can control all the six rover movement high torque motors individually. Also able to set in high torque or high speed mode. In every mode it can command the rover for 5 rank speed based on PWM. The 6 DOF rover arm is also controlled from this module. Also 2 camera’s 360 degree rotation is controlled from this module.



Figure 4: Command Input Device Selection Algorithm



Figure 5: Command Process Procedure

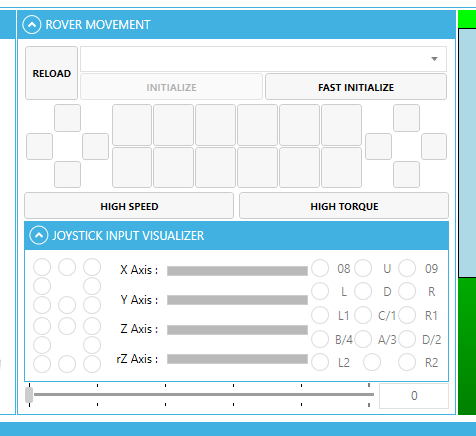


Figure 6: Rover Movement Module

# Rover Visualization Module

The primary task of this module is to visualize the rover’s current position in base. This module processes the received data string from 10 DOF sensors data which mounted in rover’s body. To do this this module takes help Connector one Module. Connector One retrieves data from Raspberry Pi and make accessible for Rover Visualization module.



Figure 7: Rover Visualization Algorithum

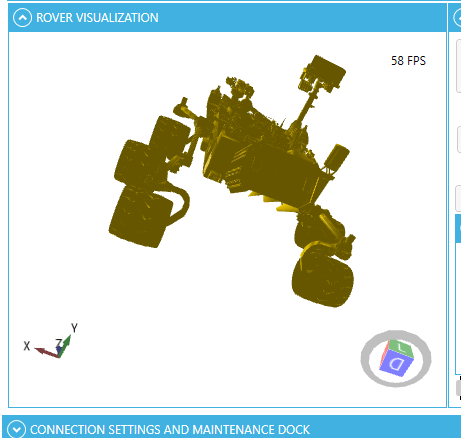


Figure 8: Rover Visualization Module

# Arm Visualization Module

Arm visualization module works in same produce as rover visualization module. Also flows the same principles.

# Web Camera View Module

This module streams the camera view in the base and shows the current situation of the rover and the sounding of the rover it receives necessary details from the settings and maintenance dock and streams the rover surroundings. Additionally it can also record the stream



Figure 9: Webcam View Module's Working Algorithm

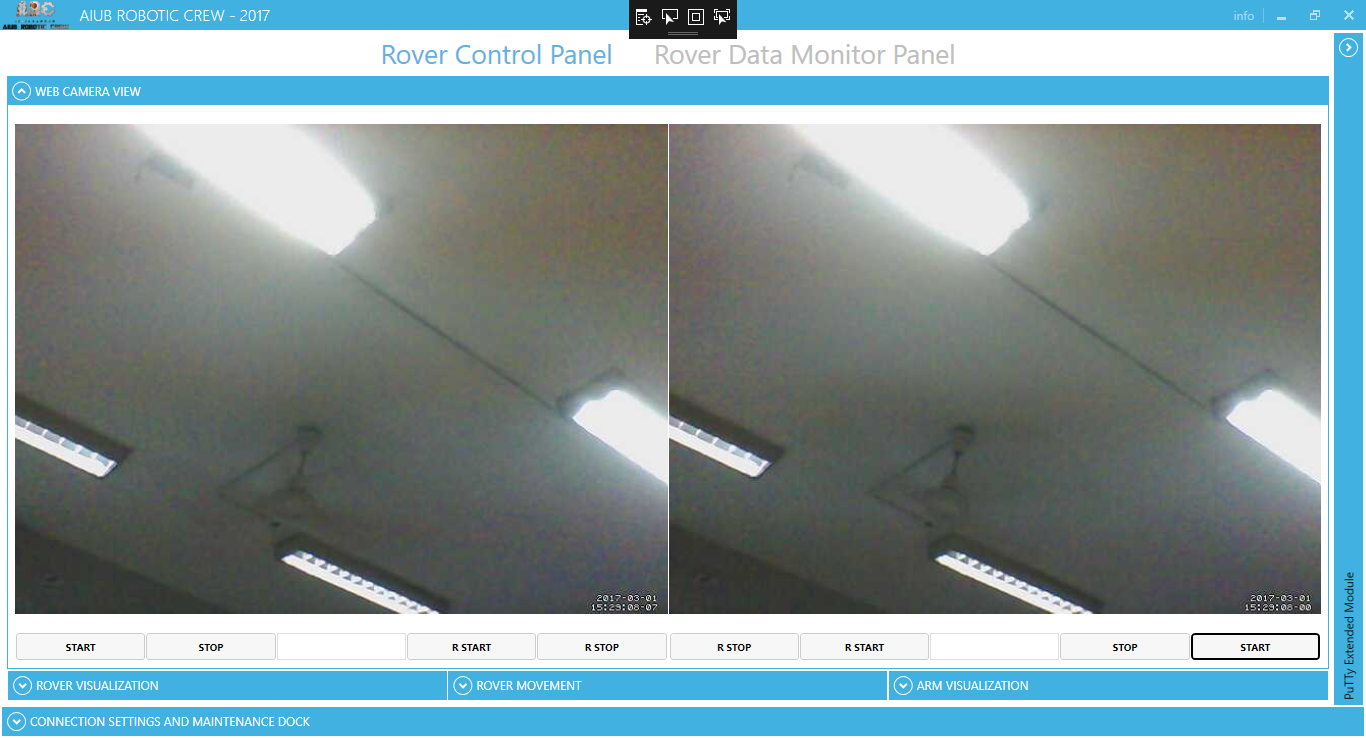


Figure 10: Webcam View Module

# Connection Settings & Maintenance Dock

This is one of the most important module in this software. It can store all the necessary details to connect with all the 3 raspberry PI’s and 2 camera also checks the connections continually is the connection is alive or not. If any of the connections gets lost this automatically provides procure to re-establish the connection. This module is also responsible for all other details keeping task. Like destination of recorded stream from rover.

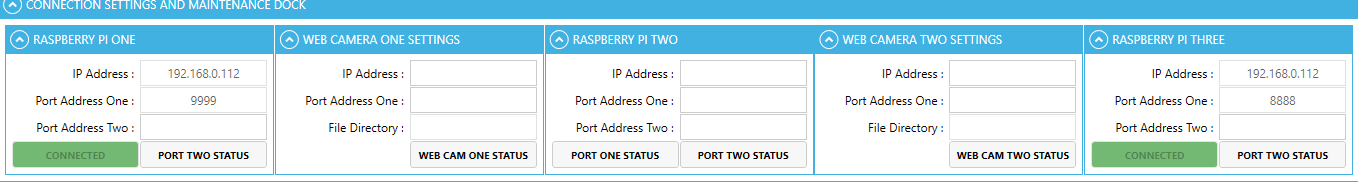


Figure 11: Settings Module

# Rover Feedback Module

This module is dedicated for receiving the feedback data from rover. First of connector one retrieves data from raspberry pi and stores it to avoid inconstancy. Then rover feedback module collects the data and processes it finally gives output to the control software. It also store data to database for future use of the data.



Figure 12: Rover Feedback Module Algorithum

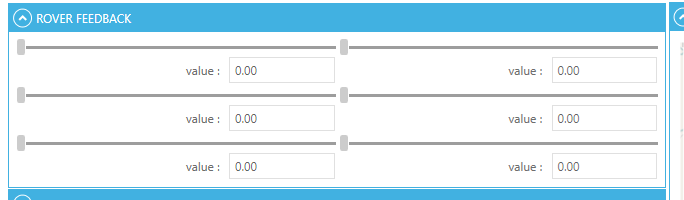


Figure 13: Rover Feedback Module

# Rover Sensor Module

Rover Sensor module Flows the same footstep of rover feedback module this time only difference is the data unless every thing is same.

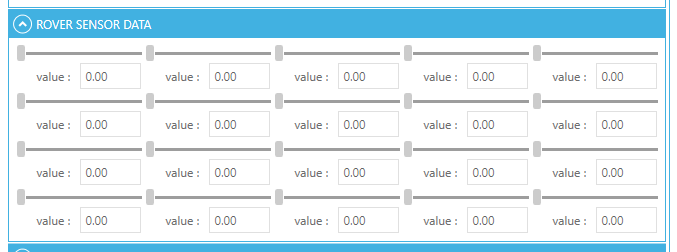


Figure 14: Rover Sensor Module

# Rover Console Module

Rover console module is one of the most Advanched tech in this control software. It has an Artificial Intelligence Build in.

The main purpose of this module is to show the details of the commands which commands are passing and executing which way. From this module Advanched operators can give command. In this module details data has been sown.

## Features

### ARC\_EYE

When any command or data is received or passed from one module to another module this feature is used.

### USER\_CONTENT

When Advanched Operator give any command to rover console in that case this feature is used.

### ARC\_EYE\_AI

If rover control software or rover console module finds any problem due to executing the command or autonomously takes any decision this feature is used.

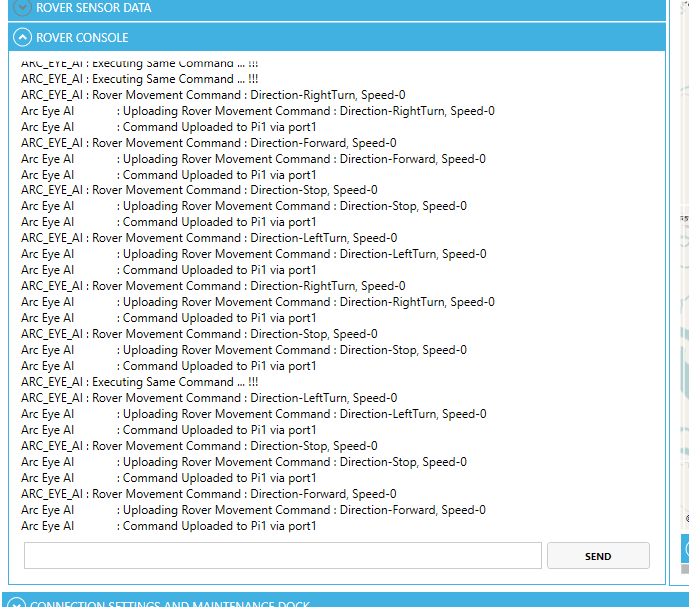


Figure 15: Rover Console Module

# GPS Navigation Module

This module is used for tracing the rover from any distance or to base station. This module Is also use forget the distance between two person or something else. By this module we can navigate our rover.

It receive data from raspberry PI then checks where the rover I gong in the right deration

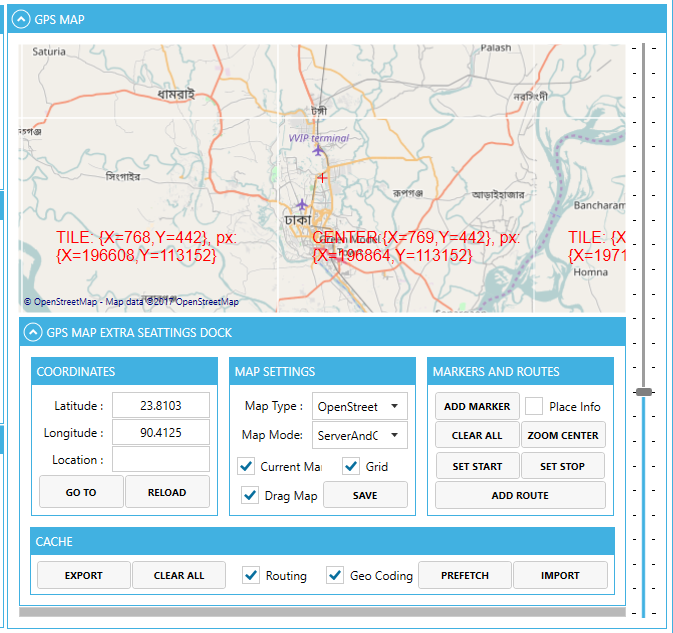


Figure 16: GPs Navigation Module

# Connector One:

This Module is responsible to communicate with all the raspberry pi and the web cam. This module also send or receives data directly from raspberry pi. This module is like the middle person between control software and raspberry pi’s



Figure 17: Connector One Module's Architecture

Thank you

Sabbir Ahmed Shourov