

# Control Software (Graphical User Interface)

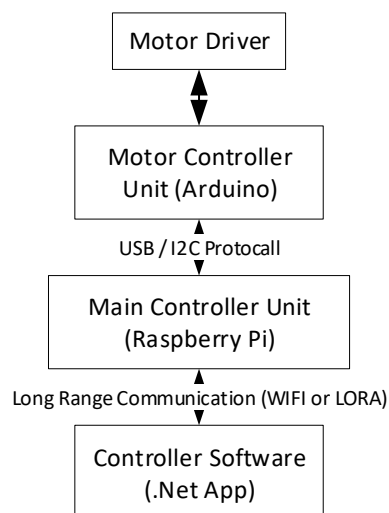
## Introduction

For the GUI development we have used the .Net platform. It was implemented by C#. the GUI consist of different modules and gauges which are assigned for different tasks.

## System breakdown

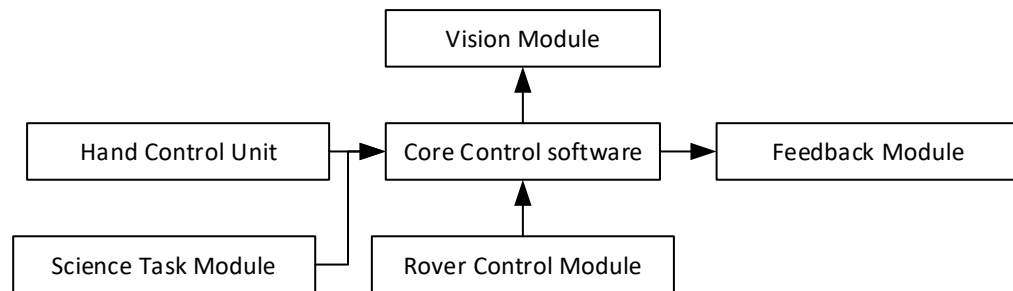
### Network Communication

This system is capable of handling large amount of data without any delay or lose of data. At first control signal was send to the main controller unit(raspberry pi) then controller unit relays the data to the motor controller unit then it transfers the data to the motor driver. This system also relays different kind of feedback data to the controller or control software. This helps the operator to take critical decisions.



### Control Software Architecture

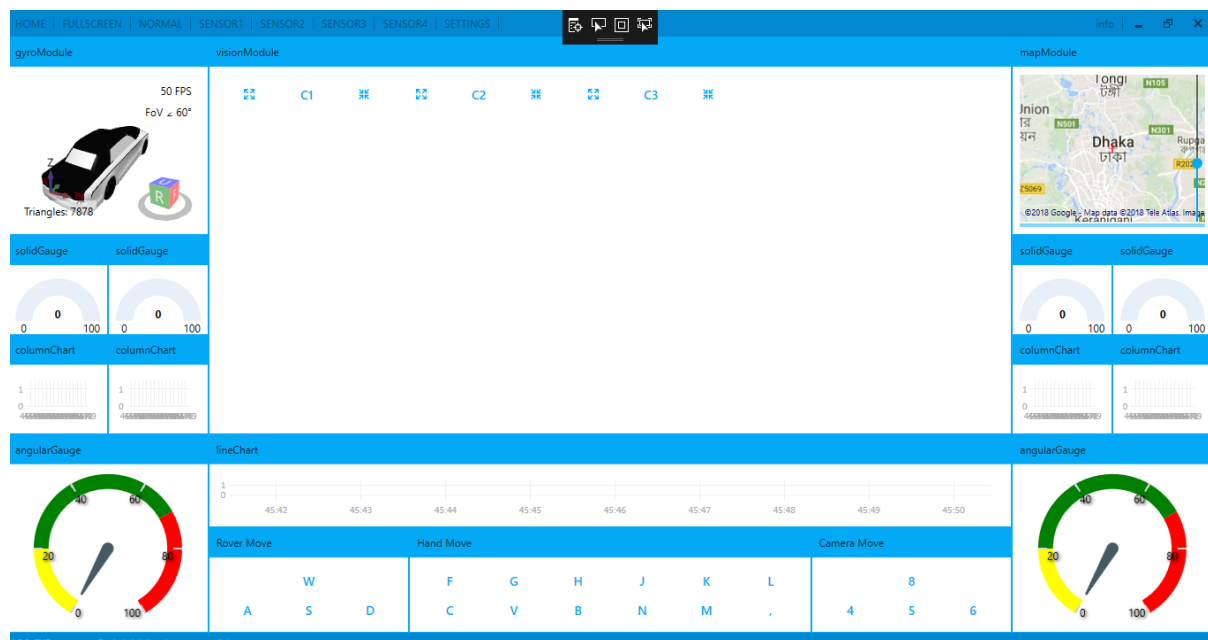
We are using state of the art software architecture. A core module is responsible for all the internal communication and outside communication (communication with raspberry pi with TCP protocol).



## Framework or libraries

- XD Messaging (IPC Framework): to make high performance communication between different modules.
- Martial Design Theme & MahApps: Used for better and complex UI design.
- Cef-Sharp: For Handling Vision interface of the rover.
- TCP Protocol: For communication with control hardware (Raspberry PI)

## Screen Shot



## Work in Progress

The Hand control module and the rover control module isn't integrated with the GUI.

## Risk Assessment

1. The system can fail due to poor signal quality or poor connection between control software and control hardware.