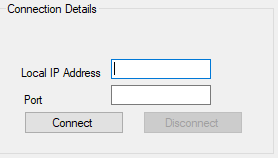
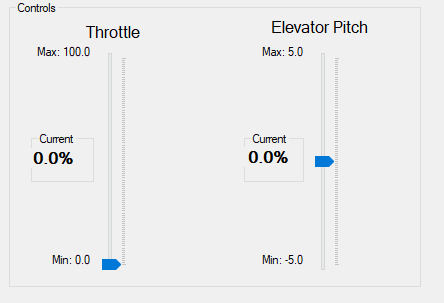
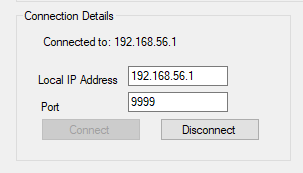
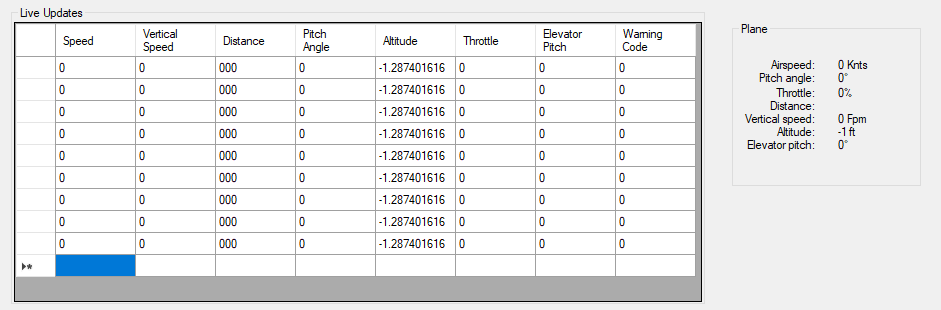
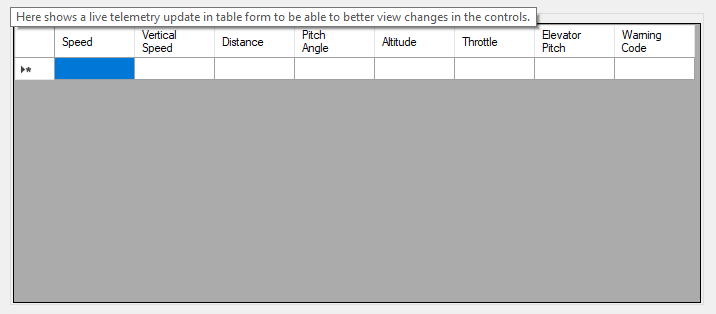
# User Guide For Remote Flight Controller



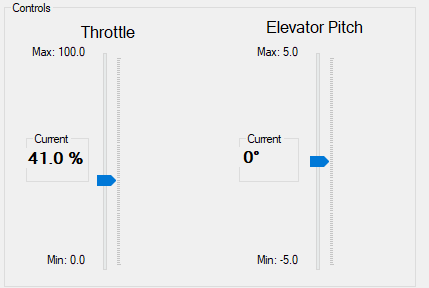
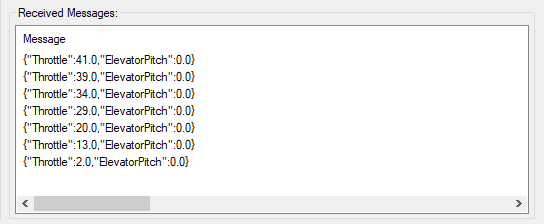
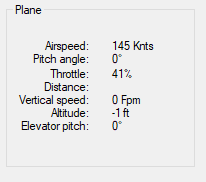
When you first load up the flight controller you are presented with the main design of the program with some disabled features. You need to first connect to the simulator with the relevant IP and port number.

Once connected a label will appear showing you are successfully connected to the IP and the flight controls will now become available



When the simulator has been started by pressing “Start Simulation” The data grid will start populating and the Telemetry updates will start coming in.

Once the simulation has started you are free to use the controls which will send updates to the simulator and the simulator will send back telemetry updates to the controller for you to view

Here you can see ne changing the throttle to 41 on controller

The simulator receives that instruction and changes its throttle

New throttle value is now displaying on the controller

## Use of Delegates

Delegates are used to pass an argument around to and from functions. I used a delegate to store the telemetry updates received from the controller and the control updates to send to the controller. This way you can send a set of instructions or receive a set of telemetry updates from anywhere in the code.



## Use of Events

Events can be used to signal other areas of the program that something has happened. In my controller I use an event for receiving data to signal that data has been received and an event for sending data to signal that data has been sent

## Use of Threading

Threading is used to run more than one piece of code at the same time. In this example we want to send control instructions to the simulator while also receive telemetry updates so we can confirm that our instructions are being carried out.

If too may threads are running this can bottleneck performance.

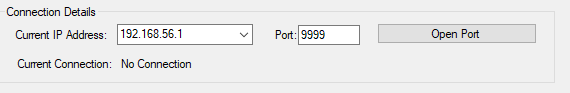


Here you can see I am starting a new thread to receive data by calling the RetrieveData function.

## Use of Sockets

Socket programming allows the programmer to develop a method of communicating over a network by using sockets. To do this we need to set a port and an IP address to be used. In this example the controller acts as a server whereas the controller acts as a client.

To allow us to connect our program to the given simulator we first need to open the port on a given IP address



Once the port has been opened we can use our program to connect to it. The sockets library will need to be included to allow this functionality.

## Use Of Serilization

As per the assignment instructions we are required to convert data into JSON format (JavaScript Object Notation) before it is sent and convert it from JSON format when received. By doing this we greatly reduce the size of the data stream and it allows better performance in the program. JSON is also readable to people, albeit a long string of text but there will be separators that the human could easily read once they know the order of the pieces of data being sent/received.

## Enhancements

I have tried to include a few different enhancements to my program. The main one being the data grid. It allows the user to clearly see each piece of data as it scrolls and updates. It is nicer to look at than a simple table and in my opinion works better.

I have added colour coded warnings so if a warning is received it will be displayed in red and if there is not warning the text will be green.

I have added tool tips in various places to explain what the control does to the user when the mouse is hovering the control.

I added a label to show when you have connected successfully to a host by displaying the connected IP address.

I started to implement and auto pilot feature but didn’t manage to get it completed

I also started to implement a disconnect feature but haven’t had the time to complete that either.