CS10220 Virtual Serial Network Analysis Report

-Jakub Jozef Grzebinoga (jag88)-

Table of Contents

Introduction ***.....2***

Challenges ***.....2***

Grade reflection ***.....3***

Introduction

This document will highlight and detail my experience of working on the Virtual Serial Network Analysis. I will be explaining the challenges that I encountered during this assignment (from the overall task to more specialised issues, such as arithmetic operations) as well as going into the solutions I came up with to overcome them. I will also highlight the tasks I have completed and reflect on the assignment from a marking perspective; giving the grade I have worked to and the reasoning behind that.

Challenges

The biggest challenge I encountered in this assignment was at the very beginning; trying to start the work for the assignment. The reasons for this were that I could not figure out the gist of the assignment and I failed to understand how the ASSP 'emulation mode 6' functioned. Starting the work was the most arduous challenge as it took me two days to overcome.
As this is not my first time working on such a project, I knew that the most important part of developing a good working solution is to fully understand the requirements, the functionality and short comings of the system the solution is to be used on. One of the methods I use to develop such an understanding is trying to abstract the problem and resources at my disposal; doing this usually allows me to see the problems from a different side than what is shown on the brief. In the case of this assignment however, I had real difficulty completing this part of the process as I did not see everything I needed to.
I tried to resolve this by myself at first, trying to put the code from my assignment into parts of the solution for worksheet 5. Unfortunately, this confused me further as the code worked in the worksheet 5 solution, but not in the assignment solution (even though it had been largely unchanged). This prompted me to speak with the lecturer that set this assignment, who explained that; the Arduino uses 4 networking ports and that it randomly assigned emulated devices to each of these ports when the emulation mode is set to '6'.
Given this information I was able to quickly understand the requirement set by each task, abstract it and work out the solution to it, which in turn allowed me to quickly complete the coding part of the assignment. Once I understood the assignment properly, completing it took around 9.5-10 hours.

The next most difficult challenge I met was formatting the output of the list of 'not connected' devices at the end of task 4. This was failing because the output was difficult to read as the device names were not space separated and were followed by random characters, even though the serial monitor was printing the names of unconnected devices successfully.
I knew the reason for this was that the serial monitor was trying to print null values at specific indexes in the **`conDevices[]`** array, however the best way to avoid this was initially unclear. I tried to resolve the problem by comparing the string at the index to '**`\0`**', no character (**`''`**) and a blank space (**`' '`**) with the **`strcmp()`** function, but this did not work.
In the end I realised that the characters in the random character output did not include separator characters, whilst all the useful strings were only three characters long. As a result, I changed the **`if()`** statement condition to filter against any string that was *not* three

characters long; after this the device name would only be printed if the string from the **`names[]`** array did *not* match the string from the **`conDevices[]`** array at the same index (i.e. if the device name was not added to the **`conDevices[]`** array during the port by port device discovery operation and was therefore not connected to any port) and if that string was exactly three characters long.
The only change that was needed after this was separating the device names with a space, I did this by printing a space using the **`Serial.print()`** function.
This issue was resolved, however it came back with no apparent reason and I had no time left to resolve it again.

The other challenges were minor in comparison to the ones mentioned above, and I was able to complete them quickly. Examples of these were; knowing when ASSP messages were successfully sent using **`SerialShield.sendASSPMessage()`** (solved by setting the verbose debug ASSP mode to true), using the correct arithmetic for task 2 (solved by working out one or two of the cases by hand to realise the two variables needed to swap places in the arithmetic operation) and understanding the difference between tasks 3 and 4 (solved by taking some time to analyse the assignment brief).

Grade reflection

I believe the work I have done is worth a high 2:1 grade (between 60% and 69%). I believe this is justified because I developed solutions to all of the tasks specified in the assignment, ensured that the code is easily readable, used defensive programming conventions, and commented the code as thoroughly as possible (ensuring that anyone who can write code can read it and understand what it does and how it does it). I also ensured that the programs serial output was formatted in the same way as the examples provided in the assignment brief, and that the user was aware of which task was being executed at the time by adding a line of output saying this.
I do not think that this is worth a higher grade than this however, as there is room for improvement. Given more time, I would add functionality that lets the user know that the program is still working, as this is not clear at times due to the time required for some tasks to finish. I would also try to make the program more efficient by putting repeated code written for debugging into modular functions, and make task 4 properly print not connected devices.

Serial Output Screenshots

Task 1 output

```
Virtual Serial Network Analysis
Doing task 1...
Device       Port3?
Ann          NO
Bug          YES
Car          NO
Day          NO
End          NO
Fax          NO
Gym          NO
Hub          YES
Ink          NO
Jam          YES
Kit          NO
Lab          NO
Mac          NO
Not          NO
Oak          YES
Pen          NO
Qua          NO
Rob          NO
Set          YES
Tea          NO
Use          NO
Van          NO
Web          NO
Xis          NO
Yes          NO
Zip          NO
```

Task 2 output

```
Virtual Serial Network Analysis
Doing task 2...
Device       Port4 Time
Ann          1250ms
Bug          NC
Car          NC
Day          NC
End          NC
Fax          NC
Gym          NC
Hub          NC
Ink          1174ms
Jam          NC
Kit          NC
Lab          NC
Mac          NC
Not          NC
Oak          NC
Pen          613ms
Qua          NC
Rob          NC
Set          NC
Tea          NC
Use          293ms
Van          851ms
Web          NC
Xis          934ms
Yes          NC
Zip          NC
```

Task 3 output

```
Virtual Serial Network Analysis
Doing task 3...
Device       Port
Ann          4
Bug          4
Car          3
Day          4
End          4
Fax          4
Gym          4
Hub          4
Ink          4
Jam          4
Kit          4
Lab          4
Mac          4
Not          4
Oak          4
Pen          4
Qua          4
Rob          4
Set          4
Tea          4
Use          4
Van          3
Web          4
Xis          4
Yes          4
Zip          4
```

Task 4 output

```
Virtual Serial Network Analysis
Doing task 4...
Port       Devices
1          Day Not Rob Tea Web Yes Zip
2          End Fax Gym Kit Lab Mac
3          Bug Hub Jam Oak Set
4          Ann Ink Pen Use Van Xis
NC
```