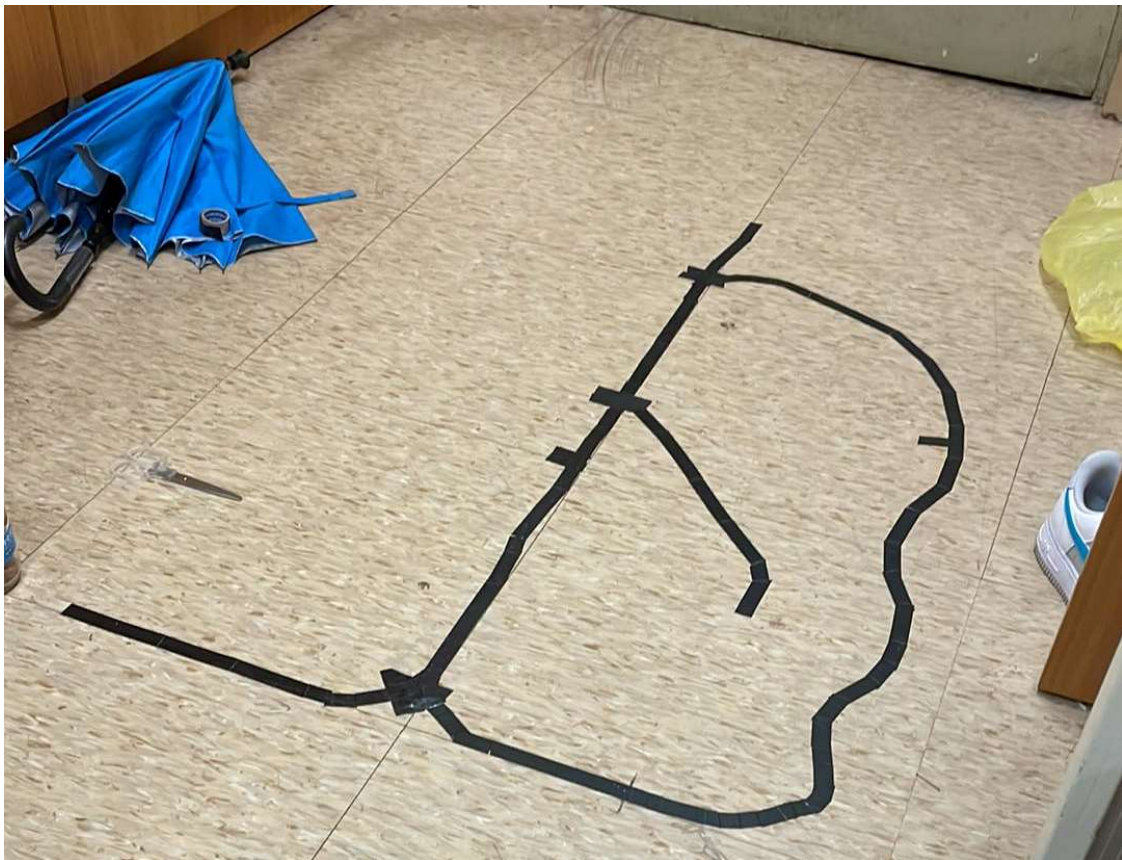


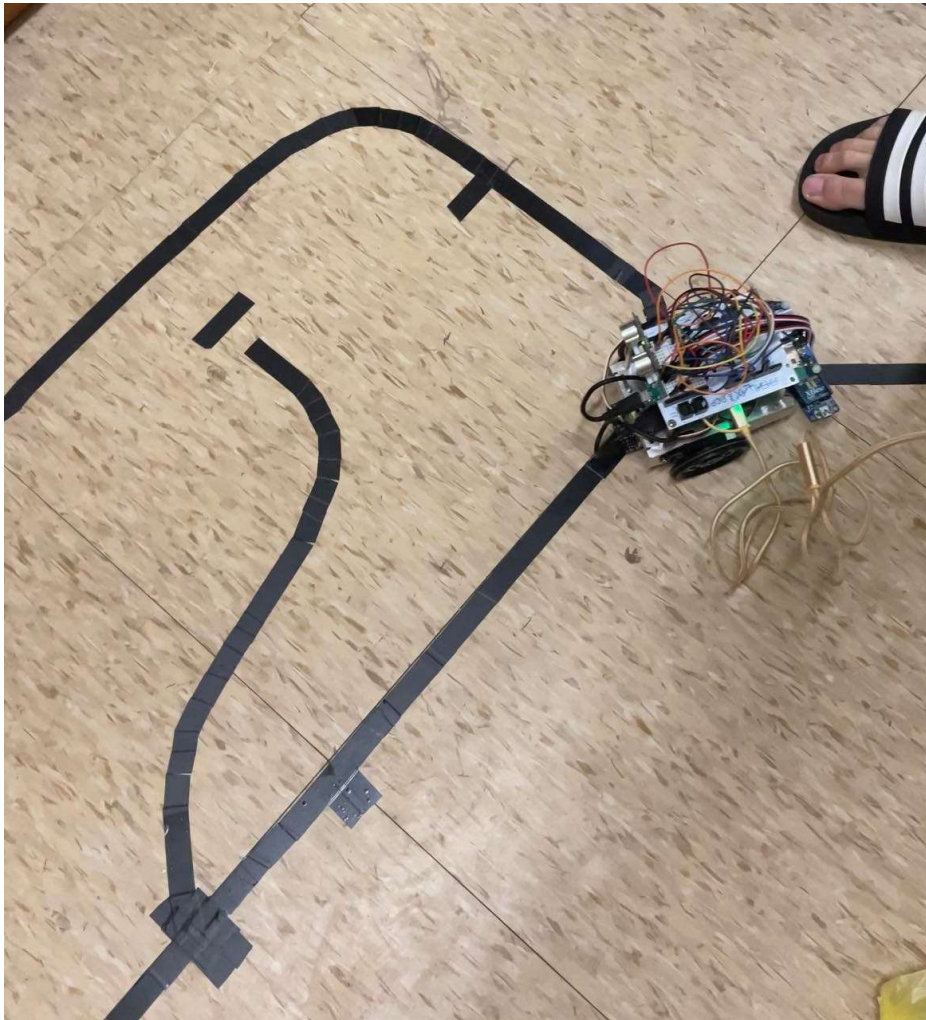
(一) 各部分步驟與結果

首先把車子組裝好之後，並把 ping, encoder 以及繁雜的接線接好後，就開始設計地圖以及寫程式了。

以下為我設計的地圖：

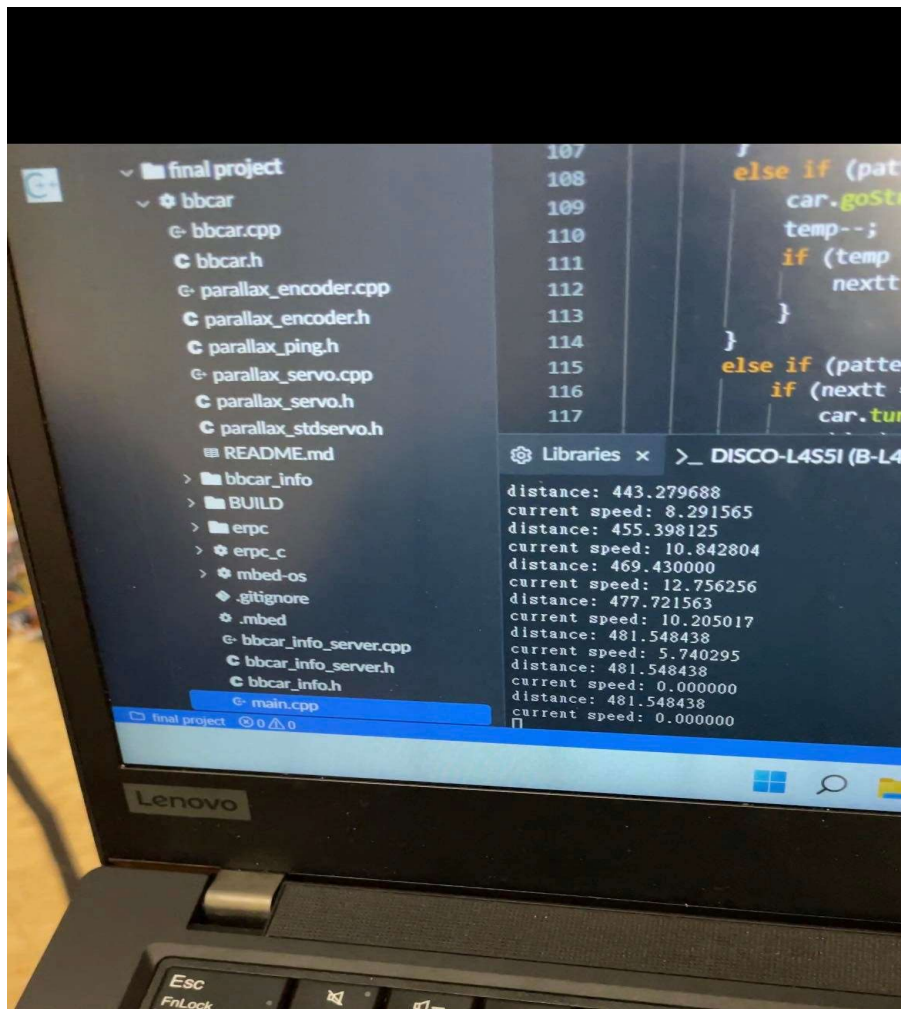


程式內容包括在偵測到 **marker** 的時候要把哪些變數轉為多少才能讓車子在下一個 **branch** (遇到 1111) 時判斷行進的方向以及遇到障礙物時要怎麼處理。我的處理方式是讓 **ping** 偵測到 20 公分內有障礙物時讓車子旋轉半圈然後原路返回。返回到上一個 **branch** 後，又會讀到相應的 **marker**，再次進到 **branch** 並選擇與剛剛不同的路。而方向向右的 **marker** 代表下一個 **branch** 要向右轉，向左貼的就是要向左轉。



至於 **erpc** 的部分，在創建完 **shim code** 之後，我在 **python** 寫了程

式，讓 erpc 不斷呼叫 mbed 會印出速度與距離的 function，而 erpc 的程式則包在一個 thread 裡面執行。速度與距離的部分，距離我就讓 encoder 來讀取並存在一個變數裡，而速度則是每秒用這次的數據減去上次的數據並除時間來得到。最後車子能完整的跑完一次賽道，並可以顯示當前的速度。



中間 Qti 的 if else 條件部分也加了許多程式來防止出問題的可能性。最後，我的車子在遇到 0000，也就是空白時會到達終點並停下來。

（二） 遇到的問題

因為這次加入的功能很多，所以常常會現不知道為什麼的錯誤，也需要不斷調整。像是 ping 跟輪子有時候無法配合、Qti 讀錯東西等錯誤。除此之外，輪子也很常脫落，到最後我不得不把輪子用螺絲鎖上。而且因為要用電腦連接板子，所以現就必須接的很長，有時候還會卡到輪子。

另一個問題是在 demo 的時候因為板子脫落，所以失誤出現的非常多，即使平常測試時沒問題且所有功能完整，在 demo 時卻還是出現了平常沒有的錯誤，覺得十分可惜。

（三） 討論

整體來說非常累，因為要不斷調整許多參數來維持車子的穩定，有時候甚至怎麼調都無法調好。最大的遺憾還是 demo 時沒有完好的 demo 完，展現原本的狀況。整體來說製作車子是很有趣的單元，雖然製作過程很累，但車子奔馳的結果會讓人覺得花的時間很值得。