



# 功能型水下遙控載具之 人機介面開發

112.08.03

Development of Human machine interface for Remotely operated underwater vehicle

著者／曾永順、林聖義、蔡渙良、黃俊誠

Zheng, Yong-Shun、Lin, Sheng-Yi、Tsai, Huan-Liang、Huang, Chun-Cheng

曾永順 大葉大學電機系  
林聖義 大葉大學電機系  
蔡渙良 大葉大學資工系  
黃俊誠 玉豐海洋科儀

## 摘要：

具有清潔的功能型水下遙控載具 (Remotely operated underwater vehicle, ROV) 是一種專門設計用於執行水下清潔任務的先進機器人技術。這些 ROV 的主要功能是清除水下結構物上的污垢、海草和海洋生物附著物，以確保結構物的運行效率和環境保護。清潔型 ROV 需要具有高效的清潔系統、靈活的運動能力和先進的影像技術，可以實現精確的清潔操作並確保結構物的運行效率和環境保護。

本研究依照 V 型開發流程來進行人機介面的設計，設計一個直觀且易於操作的 ROV 控制系統。人機介面是 ROV 操作員與機器人系統之間的主要交互界面，必須具備直觀、易於理解和操作的特點，以提高操作效率和準確性。

本研究透過 Microsoft Visual Studio，使用 C# 來開發 ROV 的人機介面。這個語言提供了豐富的圖形使用者介面 (Graphical User Interface, GUI) 工具和函式庫，可以使用按鈕、滑桿、視覺化數據顯示等，成功實現 ROV 人機介面所需的各種功能。

**關鍵字：**ROV、V 型開發流程、Microsoft Visual Studio、C#

## 壹、前言

人機介面開發主要是為了改善人與機器的交互作用 [1]，提供更好的使用者體驗和效能。功能型 ROV 的任務通常需要高度的精確度和細微的操作，設計一個直觀且易於使用的人機介面可以幫助操作人員更快速地學習和掌握 ROV 的操作，從而提高任務執行的效率。

水下清潔任務可能存在著一定的風險，例如水下結構物的不穩定性或陡峭的海床地形 [2]。通過設計人機介面，可以提供操作人員與 ROV 之間的即時數據和視覺反饋，幫助操作人員更好地了解周圍環境，避免危險情況的發生，提高操作的安全性。

功能型 ROV 通常具有多個功能和控制選項，操作人員需要在一個複雜的環境中進行操作。通過設計直觀的人機介面，可以簡化操作流程，使操作人員能夠快速選擇所需的功能並進行操作，減少出錯的可能性。

設計具有多個高解析度顯示器和即時影像傳輸的人機介面 [3]，可以讓操作人員清楚地觀察 ROV 所處的水下環境，提供清晰的數據和視覺反饋，幫助操作人員更好地理解任務情況，做出準確的決策。

## 貳、理論基礎

軟體工程是一種系統性、紀律性、量化的方法，用於軟體的開發、運營和維護；也就是將工程應用於軟體開發。本章節主要探討瀑布模型、V 型、螺旋模型、以及敏捷模型，並從中篩選最適用於功能型 ROV 之軟體開發。

### 一、瀑布模型

瀑布模型特別重視後續階段之間的相互作用，軟體測試不是嚴格遵循實施階段的活動，因此在軟體開發過程的每個階段，都必須將所獲得的結果與上一個階段進行比較，在所有階段中，都需要評估和控制品質。

瀑布模型中強調在實際建構系統之前進行詳細的分析，以免將大量精力投入到一個無法滿足用戶需求的系統中。在圖 1

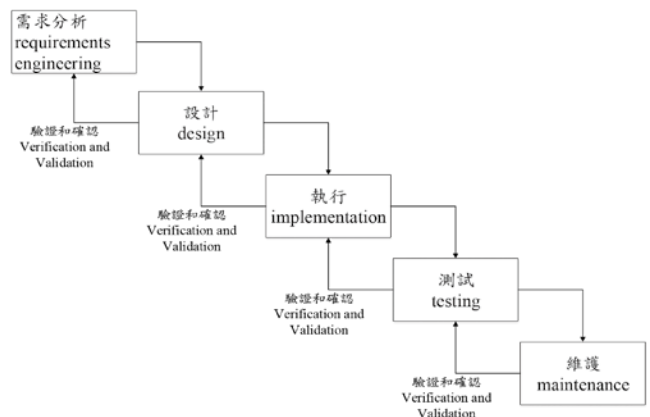


圖1、瀑布模型[4]

中，為了確保系統符合其需求並且能夠正確構建，必須評估執行到下一階段的正確性，同時驗證系統是否滿足用戶的需求，以確保在建構正確的系統。

因此，在專案早期必須清楚明確用戶的需求，才能確保正在開發的系統符合用戶期望，為了實現這一點，需求應該被詳細地記錄在需求規範文件中，在後續階段，可以藉由對需求規範文件進行驗證，確保系統的開發與用戶需求一致。

瀑布模型的重點是確保系統在開發過程中充分滿足用戶需求，並通過仔細的分析和需求驗證來實現這一目標，以避免投入無法滿足用戶需求的不必要工作。

## 二、螺旋模型

軟體工程的螺旋模型（如圖 2）已經進化了幾年，基於對瀑布模型在大型軟體開發項目中的各種改進的經驗，螺旋模型可以容納大多數以前的模型作為特殊情況，並進一步提供哪種組合最適合目前軟體的開發。

在圖 2 中，半徑維度代表截至目前為止完成步驟所產生的累積成本；角度維度代表完成每個螺旋週期所取得的進展。該模型顯示了一個重要的概念，即每個週期都包含一個進程，從整體運營的設計文檔到

每個單獨程式的編碼，對產品的每個周期各個部分進行相同的步驟。

螺旋模型的每個週期始於以下的辨識：

- 正在詳細說明的產品部分的目標（性能、功能、適應變化的能力等）。
- 實施該產品部分的替代方案（設計 A、設計 B、重複使用、購買等）。
- 對應用替代方案所施加的限制（成本、進度、介面等）。

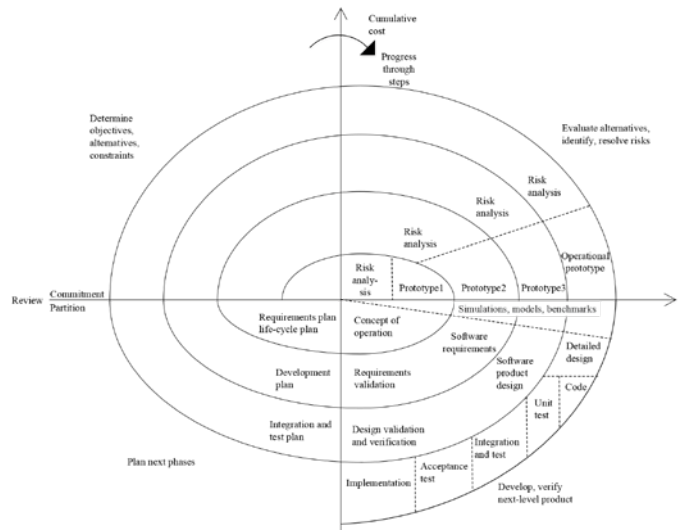


圖2、螺旋模型[5]

## 三、V 型

V 型模型是對瀑布模型的一種變異，其表示方式稱為 V 型。在圖 3 中呈現的 V 型模型，描述了與通信、建模和早期構建活動相關的品質保證行動之間的關係。

當軟體團隊沿著 V 型左側向下移動時，基本的需求被逐步細化為更詳細和技術性

的問題及其解決方案的表示。當開始進行編碼後，團隊就會沿著 V 型的右側向上移動，進行一系列的測試（品質保證行動），驗證軟體在沿著左側向下移動時創建的每個模型。事實上，傳統的生命周期和 V 型模型之間沒有根本性的區別，V 型模型提供了一種將確認和驗證應用於早期工程工作的可視化方式。

- 個人和互動優於流程和工具。
- 能運作的軟體優於詳盡的文件。
- 與客戶的合作優於合同的洽談。
- 對變化的回應優於遵循計劃。
- 敏捷模型重視個人和互動勝於流程和工具，如圖 4，因為他們知道，一個出色的軟體是由優秀個體共同創造的，激發每個成員的潛力並促進彼此之間的有效溝通和合作時，才能產生最佳的結果。這種模式驅動著敏捷團隊在不斷挑戰中成長和發展，確保他們能夠交付出最能滿足客戶需求的軟體。

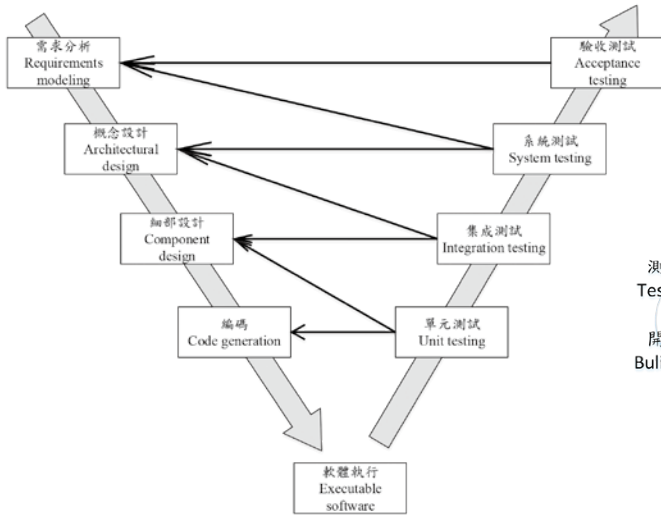


圖3、V型[6]

#### 四、敏捷模型

敏捷運動在之前已經開始了幾年，正式始於 2001 年 2 月的《敏捷宣言》（Beck 2001）的創立。當時被稱為「輕量級方法學家」的十七位專家共同編寫和簽署了這份宣言，這份文件不僅為他們所從事的工作命名，還提供了一系列價值聲明。《敏捷宣言》的作者們寫道，他們重視以下價值 [7]：

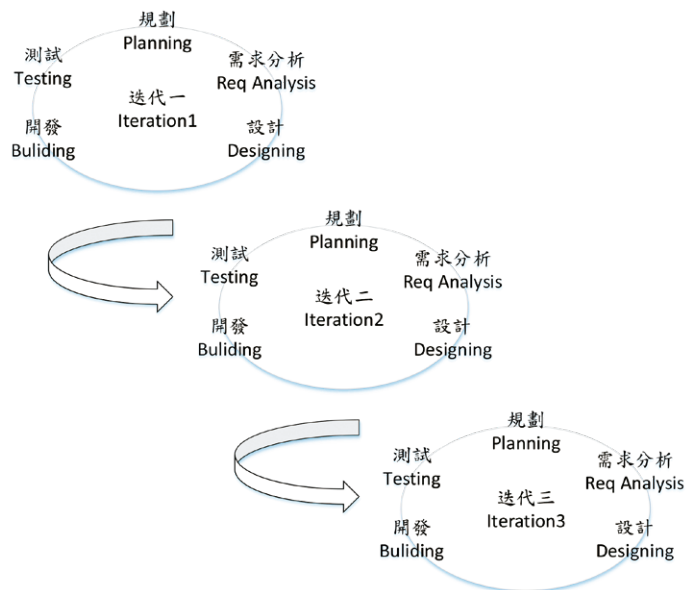


圖4、敏捷模型[8]

#### 五、本研究人機介面開發流程

本研究為小型專案，有明確的需求並且更改的機率極低，不需要跟客戶持續互

動，需要大量的測試來保證軟體的可行性，因此使用V型作為本研究之開發流程。

V型開發流程在開發前要求明確定義業務需求和系統需求，這與本研究具有明確需求的特點相吻合。由於更改的機率極低，V型開發流程的結構化開發和測試方法非常適用。在圖5中，針對專案的製作流程，根據設計階段的不同，分為需求分析、功能分析、概念設計和詳細設計等階段。這種逐步推進的方式有助於在每個階段都進行相應的測試活動，以確保質量和穩定性。

在測試階段，V型開發流程也提供了明確的測試階段，本研究將測試階段分為訊號測試、各部件測試、功能測試和驗收測試，這與V型開發流程中測試階段的對應相符合，通過在各個階段進行相應的測試活動，可以及早發現和解決問題，減少後期修改成本。

## 參、人機介面設計階段

本階段流程分為需求分析、功能分析、概念設計、詳細設計，在需求分析階段先收集並分析系統的需求，在功能分析階段使用需求分析的結果來進行功能分析，概念設計階段是在需求和功能分析的基礎上，進一步定義系統的整體設計方案，在詳細設計階段，進一步細化概念設計的結果。

### 一、需求分析

本載具要在水下清潔海生物，需具備在水下自由活動和清潔的能力，根據功能需求報告書的系統需求，需要以下裝備：

- (一) 七組推進器：兩組水平推進系統、兩組垂直推進系統、兩組履帶系統、一組任務系統。
- (二) 兩組 LED 燈。
- (三) 兩組攝影機 (IP camera)。
- (四) 一組 PCG (深度計、指南針、陀螺儀)。

### 二、功能分析

本載具人機介面須具備以下功能：

- (一) 各軸推進系統與任務系統之電源開關 (需要有全開、全關與個別開關功能)。
- (二) 飛行模式可操控載具前進、後退、左轉、右轉、上升、下降、左翻、右翻。
- (三) 履帶模式可操控前進、後退、左轉、

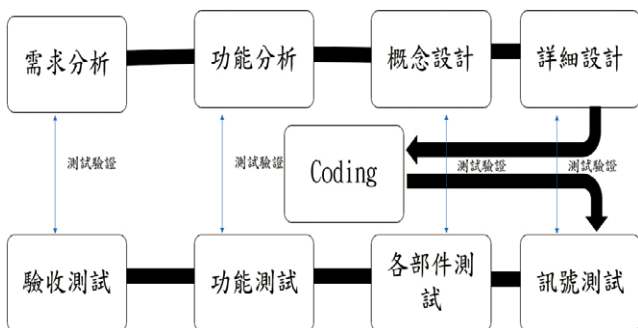


圖5、本研究之V型開發流程

右轉。

(四) 可調節各軸推進器以及刷子馬達之轉速。

(五) 可調節 LED 之亮度。

(六) 具有影像回傳及錄影功能。

(七) PCG 之數據回傳，並將載具姿態顯示在介面上。

(八) 可進行定深、定航向、及翻轉角度之控制。

(九) 翻轉後之吸附控制。

### 三、概念設計

利用 Microsoft Visual Studio 中的 Windows Form App 專案使用 C# 設計人機介面，主要使用工具箱中的 Button (按鈕) [9]、TrackBar (滑桿) [10]、Label (標籤) [11]、TextBox (文字方塊) [12] 進行設計，按鈕為控制 ROV 之方向，滑桿為控制推進器轉速以及 LED 燈亮度、標籤可自定義內部文字，通常用於說明特定資訊與資料回傳之數值、文字方塊為顯示滑桿之數值以及控制定深、定航向、翻轉角度之數值。

儀器主要利用 RS232、RS485 傳輸與接收訊號，在進行訊號傳輸與接收前需要設定鮑率 (Baud)、同位檢查 (Parity Check) 和停止位 (Stop Bit)，如表 1。

表1、各儀器使用訊號

儀器	訊號	鮑率	同位檢查	停止位
LED	RS232轉 RS485	115200	None	1
刷子馬達	RS232轉 RS485	115200	None	1
Thruster15V 電源開關	RS232	9600	None	1
水平、垂直、 履帶推進器	RS485	115200	None	1
PCG	RS232	115200	None	1

### 四、詳細設計

此章節主要講述人機介面編碼以及設計，圖 6 為程式流程圖，依照此流程進行編碼，程式主要利用以下邏輯：

(一) 載具運動步驟：

1. 滑桿調整推進器轉速。
2. 按鈕控制載具方向。

(二) LED 燈亮度：滑桿控制。

(三) PCG:接收感測器訊號顯示在標籤上。

(四) 刷子轉速：滑桿控制。

(五) 攝影機：

1. 抓取影像到介面。
2. 按下開始錄影可選擇檔案儲存位置。

根據前面的設計階段流程開發出本研究的人機介面，如圖 7 所示，主要利用滑鼠進行控制，當使用者持續按住控制 ROV 方向的按鈕時，可讓推進器持續運轉，放開按鈕即停止，滑桿可以控制 LED 燈亮

度及推進器轉速，可回傳 PCG 數據，將即時姿態以及攝影機影像顯示在介面上。本研究目標是開發簡易及直觀的控制系

統，通過直觀的人機介面，可以簡化操作流程，使操作人員能夠快速選擇所需的功能並進行操作，減少出錯的可能性。

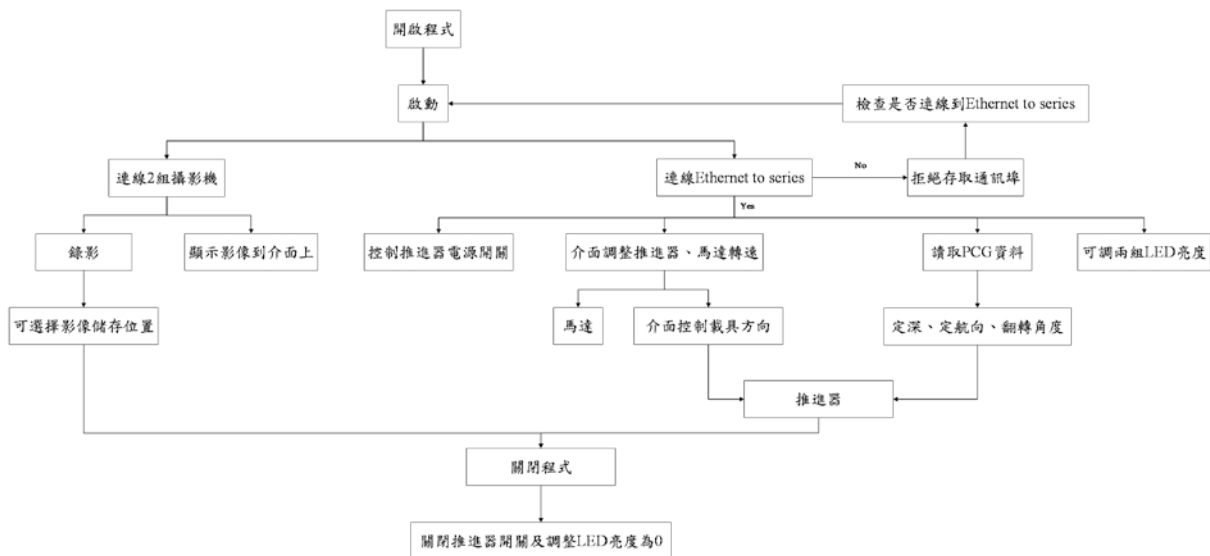


圖6、程式流程圖

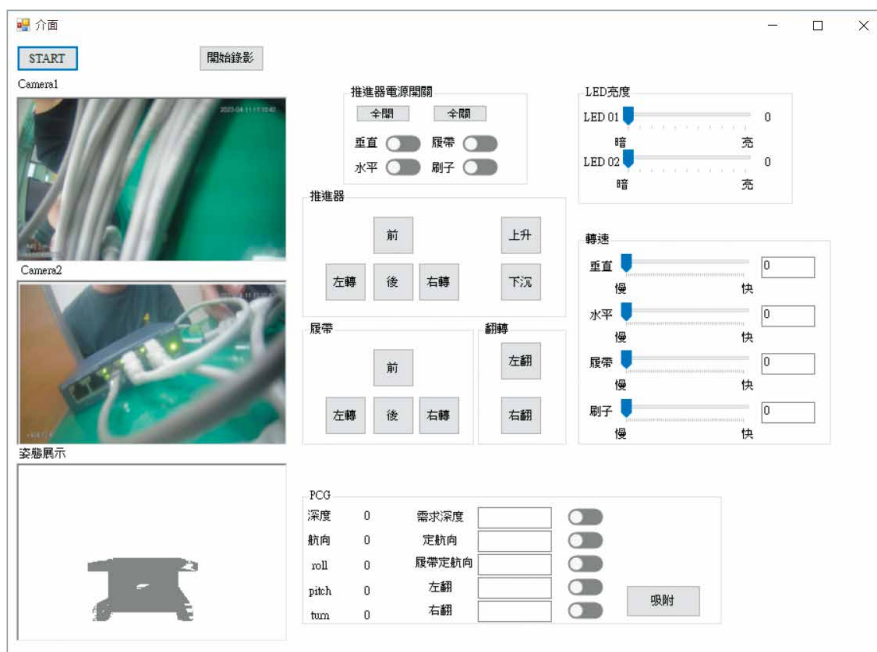


圖7、人機介面

## 肆、人機介面測試階段

本階段流程分為訊號測試、各部件測試、功能測試、驗收測試，在訊號測試階段，對系統的各個輸入和輸出進行測試，以確定是否正確地接收和傳遞訊號，各部件測試階段專注於測試系統的各個儀器，以確保它們能夠獨立運作並達到預期的功能，功能測試是在系統開發過程中的一個重要階段，在驗證系統是否符合需求並按照預期功能運作，驗收測試是軟體開發的最後階段，用於確定系統是否滿足最終使用者的需求和期望。

### 一、訊號測試

編碼完成後進行人機介面訊號短路測試，測試各按鈕及開關是否正確傳遞與回傳訊號，表 2 為訊號測試所排定之測試項目。

### 二、各部件測試

使用人機介面測試各部件功能是否可以正常運作，避免拿到無法使用以及運作異常的部件，表 3 為各部件測試所排定測試項目。

### 三、功能測試

接上電路後開始進行功能測試，利用人機介面檢測按鈕使否可以正常操作載具移動，表 4 為功能測試所排定的測試項目。

表2、訊號測試

人機介面控制項	測試項目
LED1	滑桿傳送與回傳資料一致
LED2	滑桿傳送與回傳資料一致
IPcamera1	影像回傳及錄影
IPcamera2	影像回傳及錄影
Thruster15V電源開關	開關按鈕傳送資料
推進器控制載具前進	傳送訊號與回傳一致
推進器控制載具後退	傳送訊號與回傳一致
推進器控制載具左轉	傳送訊號與回傳一致
推進器控制載具右轉	傳送訊號與回傳一致
履帶控制載具前進	傳送訊號與回傳一致
履帶控制載具後退	傳送訊號與回傳一致
履帶控制載具左轉	傳送訊號與回傳一致
履帶控制載具右轉	傳送訊號與回傳一致
推進器控制載具上升	傳送訊號與回傳一致
推進器控制載具下沉	傳送訊號與回傳一致
推進器控制載具左翻	傳送訊號與回傳一致
推進器控制載具右翻	傳送訊號與回傳一致
刷子馬達	傳送訊號與回傳一致
PCG	讀取資料
定深、定航向及控制翻轉角度	傳送訊號與回傳一致
吸附	傳送訊號與回傳一致

表3、各部件測試

各部件	測試項目
LED 1	可調節亮度
LED 2	可調節亮度
IP camera 1	影像回傳及錄影
IP camera 2	影像回傳及錄影
Thruster15V電源開關	可全開/全關及個別開關
Thruster 61(L1)	正反轉及轉速控制
Thruster 62(L2)	正反轉及轉速控制
Thruster 63(V1)	正反轉及轉速控制
Thruster 64(V2)	正反轉及轉速控制
Thruster 65(CP)	正反轉及轉速控制
Thruster 66(CS)	正反轉及轉速控制
Thruster 67(B)	轉速控制
PCG	讀取資料



表4、功能測試

裝備與元件	測試項目
推進器(飛行模式)	共四組，操控載具前進後退、上升下降、左轉右轉、左翻轉右翻轉之移動方式。
推進器(履帶模式)	共兩組，操控前進後退以及左右轉之移動。
推進器(刷子模式)	共一組，操控刷子轉動。
23W LED	共兩組，可以使用滑桿調節燈光的亮度。
Thruster15V電源開關	可以開啟/關閉"垂直"、"水平"、"履帶"、"刷子"的推進器。
IP camera	共兩組，接上電源與電腦後，可看到影像畫面。
PCG	共一組，接上電源與電腦後，可看到資料。
各軸向推進器的轉速	可以使用滑桿調節推進器的轉速。
姿態展示	讀取PCG數值後展示即時的載具狀態。

#### 四、驗收測試

主要在做廠內測試 (Factory Acceptance Test, FAT)，以利後續的港試 (Harbor Acceptance Test, HAT) 以及海試 (Sea Acceptance Test, SAT)，表 5 為廠內測試所排定的測試項目。

#### 伍、結論

本研究主要在開發功能型 ROV 的人機介面，通過設計直觀的人機介面，以簡化操作流程，使操作人員能夠快速選擇所需的功能並進行操作，減少出錯的可能性。

利用 V 型軟體開發流程的結構性特點，

表5、廠內測試

裝備與元件	測試項目
推進器(飛行模式)	共四組，操控載具前進後退、上升下降、左轉右轉、左翻轉右翻轉之移動方式。
推進器(履帶模式)	共兩組，操控前進後退以及左右轉之移動。
推進器(刷子模式)	共一組，操控刷子轉動。
23W LED	共兩組，可以使用滑桿調節燈光的亮度。
Thruster15V 電源開關	可以開啟/關閉"垂直"、"水平"、"履帶"、"刷子"的推進器。
IP camera	共兩組，接上電源與電腦後，可看到影像畫面。
PCG	共一組，接上電源與電腦後，可看到資料以及即時的載具姿態。
各軸向推進器的轉速	可以使用滑桿調節推進器的轉速。
Auto Heading 定向	啟動auto heading功能後，操作ROV轉向，ROV會自動轉回原本設定的方向。
Auto Depth 定深	啟動auto depth功能後，外部推動ROV，會自動移動回原本設定的深度(公差： $\pm 0.2$ meter)。
Auto roll 自動翻轉	啟動auto roll功能後，設定翻轉的角度( $\pm 180^\circ$ )，ROV會自動轉到設定的角度(公差： $\pm 5^\circ$ )。
吸附	關閉Auto roll 功能，並以最大推力吸附結構體。

可以有系統地進行軟體開發過程，確保每個階段的需求、設計、編碼和測試都得到徹底驗證，這使得人機介面在開發過程中快速識別和解決問題，提高品質和可靠性。

通過本研究的工作，成功實現了一個功能完整且具有良好用戶體驗的人機介面，該介面符合用戶需求，提供了直觀的操作界面、清晰的資訊展示和友好的反饋機制，這將有助於提高系統的可用性和使用者滿意度。

綜上所述，本研究運用 V 型軟體開發流程和 Microsoft Visual Studio 成功進行人機介面的編碼工作，並取得了具體的成果和貢獻。這項研究不僅提供了一個實用的軟體開發方法，同時也為未來相關領域的研究和實踐提供了參考和啟示，這些成果將對改進軟體開發流程和提升人機介面的設計和開發質量有著重要的影響。🏆

## 參考文獻

- [1] I. Qasim, M. W. Anwar, F. Azam, H. Tufail, W. H. Butt and M. N. Zafar, "A Model-Driven Mobile HMI Framework (MMHF) for Industrial Control Systems," (IEEE, 2020) , pp. 10827-10846
- [2] H. Yao, H. Wang, Y. Li, Y. Wang and C. Han, "Research on Unmanned Underwater Vehicle Threat Assessment," (IEEE, 2019) , vol. 7, pp. 11387-11396
- [3] Robert D. Christ and Robert L. Wernli, Sr. "The ROV Manual" A User Guide for Remotely Operated Vehicles Book ,Second Edition, 2014, PP. 102
- [4] Hans van Vliet, Software Engineering: Principles and Practice, (Wiley, 2007) , PP. 48-50
- [5] B. W. Boehm, "A spiral model of software development and enhancement," (IEEE,1988) , pp. 61-72
- [6] Roger S. Pre, Software Engineering: A Practitioner's Approach, (McGraw-Hill Education , 2009) , PP. 40
- [7] Mike Cohn, Agile Estimating and Planning, (Prentice Hall, 2005) , PP. 21-22
- [8] SDLC (Software Development Life Cycle) - Agile Model, [https://www.tutorialspoint.com/sdlc/sdlc\\_agile\\_model.htm](https://www.tutorialspoint.com/sdlc/sdlc_agile_model.htm) (檢索日期：2023年6月24日)
- [9] Button 類別(System.Windows.Forms) - Microsoft Learn, <https://learn.microsoft.com/zh-tw/dotnet/api/system.windows.forms.button?view=windowsdesktop-7.0> (檢索日期：2023年7月24日)
- [10] TrackBar 類別(System.Windows.Forms) - Microsoft Learn, <https://learn.microsoft.com/zh-tw/dotnet/api/system.windows.forms.trackbar?view=windowsdesktop-7.0> (檢索日期：2023年7月24日)
- [11] Label 類別(System.Windows.Forms) - Microsoft Learn, <https://learn.microsoft.com/zh-tw/dotnet/api/system.windows.forms.label?view=windowsdesktop-7.0> (檢索日期：2023年7月24日)
- [12] TextBox 類別(System.Windows.Forms) - Microsoft Learn, <https://learn.microsoft.com/zh-tw/dotnet/api/system.windows.forms.textbox?view=windowsdesktop-7.0> (檢索日期：2023年7月24日)