

基於 LLM 大語言模型與 ESP32 微控制器之 AIoT 跨領域專題課程：AI 語音智慧小車

何仁偉

國立臺灣師範大學附屬高級中學生活科技科教師

前言

隨著 108 課程綱要之推行，科技領域強調培養學生之「運算思維」(computational thinking)、「設計思考」(design thinking) 與「問題解決能力」(教育部，2018)。過往科技教育多聚焦於傳統機電整合、電路實作與基礎程式控制；然而，在全球邁入物聯網 (IoT) 與人工智慧 (AI) 深度融合之 AIoT (智慧物聯網) 時代，設備與載具已不再僅是被動接收指令的「執行器」，而是演化為具備感知、語意理解與邊緣決策能力之智慧系統 (陳侑成，2022)。

為回應此一科技變革趨勢，中學階段之科技專題實作應突破過往單一硬體開關控制之局限，轉而透過 AIoT 技術促進軟硬體系統與智慧載具之深度整合，以期培養學生具備因應未來科技所需之跨領域系統整合實務與人機互動設計的核心素養。

回顧 Arduino/ESP32 微控制器教學之瓶頸與挑戰

審視過往中小學之微控制器專題 (如 Arduino 或 ESP32 遙控載具)，其教材設計多屬於「硬體導向」為範例。學生主要透過實體按鍵、紅外線感測器、超音波感測器或藍牙 App 進行載具控制，此類專題模式雖能奠定學生在基礎電路與程序邏輯的基礎，然而在控制機制上多受限於「一對一」的僵化指令，難以因應真實情境中多元且複雜的人機互動需求。

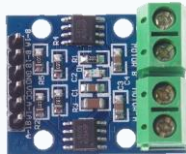
在生成式人工智慧 (generative AI) 與大語言模型 (large language model, LLM) 技術迅速普及之當代，傳統專題教學模式顯然無法引導學生體驗從「語法精確匹配」跨越至「語意容錯與廣義理解」之技術轉移，導致學生對邊緣載具控制之認知仍停留在傳統被動式遙控之層次。因此，教學模式亟需從「單一語法指令導向」轉型為「多元語意控制架構」，亦即由 AI 負責高階且複雜之語意意圖解析，並由邊緣微控制器執行精確之機電控制，以實現高度人性化之人機互動機制。

課程架構與機電整合設計

基於前述之教學背景與轉型需求，作者曾設計並實作一套具備語音指令辨識能力之「AI 語音智慧小車」專題。系統採用 web-based 網頁平台架構，學生可透過網頁介面即時進行語音輸入並進行載具控制。在硬體架構方面，本專題參考「START！智慧小車」之底盤設計，選用 ESP32 微控制器作為核心邊緣控制主板，並整合 L9110S 馬達驅動模組與 TT 直流減速馬達，打造低成本且高度擴充之機電控制系統。本專題所使用之微控制器、驅動模組與相關機電零組件規格摘要如表 1 所示。

表 1

AI 語音智慧小車相關機電零組件規格表

名稱	圖示	數量
ESP32 (Wemos D1 R32)		1 個
L9110S 馬達驅動板		1 個
TT 直流減速馬達 (1:120)		2 個
TT 直流減速馬達車輪		2 個
牛眼輪		1 個
杜邦線 (公母)		6 條
杜邦線 (公公)		4 條

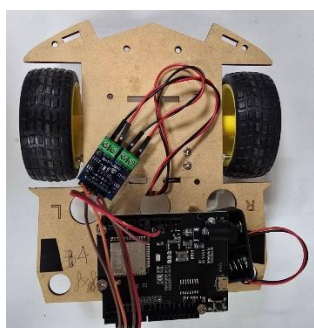
名稱	圖示	數量
M3 六角雙母細牙銅柱 20mm		6 個

將上述元件組裝於雷射切割木質底盤後，整體「AI 語音智慧小車」之硬體結構如圖 1 所示。為完整呈現小車之空間佈局與機電配置，圖 1 分別由三個不同視角展示其車體結構；圖 2 呈現 AI 語音小車實際運作影片的 QR Code。

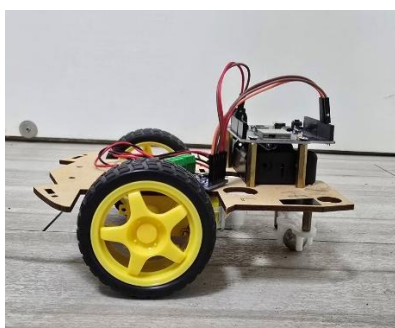
圖 1

AI 語音小車教材示意圖

(a) 俯視圖 (機電布局)



(b) 側視圖 (底盤與結構)



(c) 立體圖

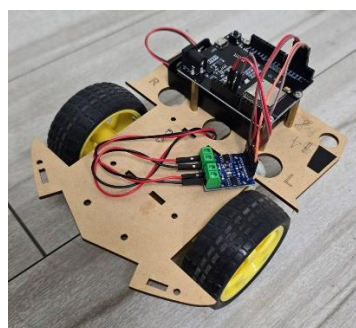


圖 2

AI 語音小車實際運行影片



註：請掃描 QR Code 或複製連結觀看完整演示影片

<https://youtube.com/shorts/ttLtHW5MnRI>

基於 LLM 與 API 通訊協定之結合

在控制介面設計上，本系統採用 HTML 架構開發 web-based 互動平台，整合「語音辨識」、「手勢辨識」與「鍵盤遙控」三種多模態控制模式。以下針對本教材「語音辨識」技術之架構與實務應用進行深入說明。

在語音辨識模式下，使用者僅需啟用麥克風裝置進行語音輸入，系統於擷取音訊並解析意圖後，即發送相應指令驅動智慧小車執行特定動作。相較於傳統僅能匹配固定關鍵字之語音控制，本系統導入 LLM 大語言模型之自然語言理解機制，將單一控制動作延伸至多維度的同義詞彙庫。例如，將「前進」指令之語意對映涵蓋「向前」與「直走」等自然口語表達；「後退」指令則能同義識別「倒車」與「退後」等口語訴求。此一 LLM 技術之整合，顯著提升了人機互動之容錯性與自然度，讓使用者得以透過多元化的語言習慣靈活驅動載具。

本教材導入 LLM 技術，結合網頁 API（application programming interface/應用程式介面）通訊協定，實現自然口語指令即時辨識，並將控制命令回傳至 ESP32 微控制器。在語音辨識之技術流程設計上，主要涵蓋「語音訊號擷取」、「語意理解與模糊匹配」與「API 資料傳輸」三階段，以下針對各階段之機制進行深入探討：

一、語音訊號擷取 (Voice Signal Capture)

第一階段使用 JavaScript 語法中的 Speech Recognition 指令，即時呼叫使用者端裝置之麥克風進行音訊擷取。系統將使用者輸入之語音訊號轉譯為繁體中文文字字串，並同步呈現於網頁控制平台。如圖 3 所示。

圖 3
網頁端語音辨識頁面

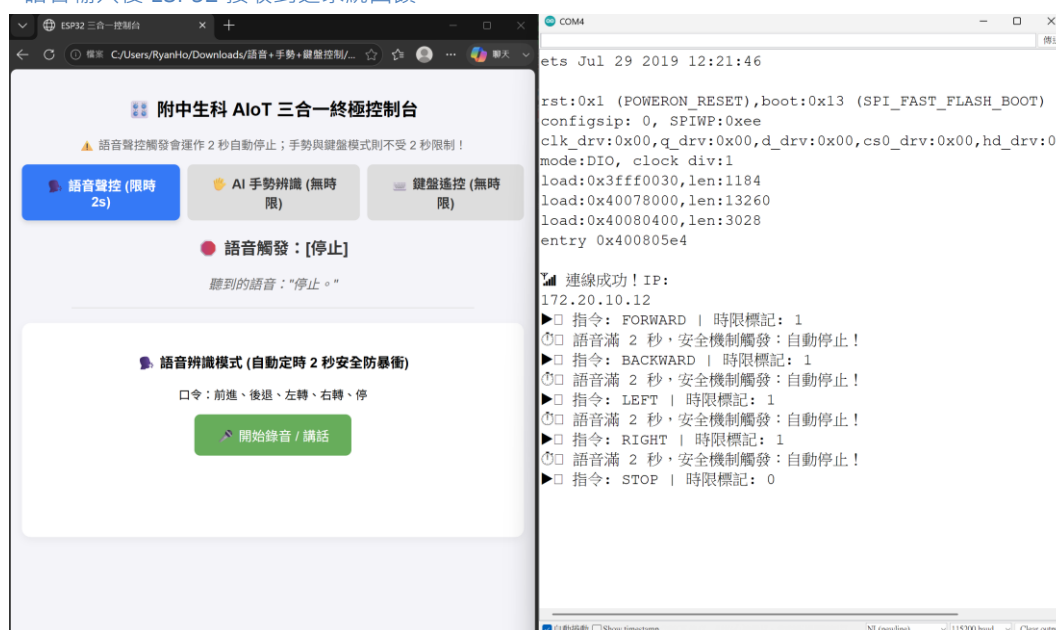


二、語意理解與模糊匹配 (Semantic Understanding & Fuzzy Matching)

第二階段將透過條件判斷式 includes () 建立「一對多」之同義詞對應機制，進而實現自然語言理解。例如：當系統辨識到使用者說出「前進」、「向前」或「直走」等同義詞時，系統均統一對應為 FORWARD 動作；若說出「倒車」、「後退」或「退後」，則對應為 BACKWARD 動作，有效提升系統對口語表達之相容性與容錯度。語音指令回傳 ESP32 微控制器之結果如圖 4 所示。

圖 4

語音輸入後 ESP32 接收到之系統回饋



三、API 資料傳送 (API Data Transmission)

第三階段當系統將語意解析完成後，系統即呼叫 sendMoveCommand () 函式，透過 HTTP API 以非同步請求方式將控制指令發送至 ESP32 微控制器，進一步驅動 AI 語音小車執行對應之行駛動作。語音辨識語法如圖 5 所示。

圖 5

HTML 之語音辨識語法與同義詞指令解析

```
// ===== 語音辨識 (帶有 2 秒定時標籤 time=1) =====
let recognition; let isRecording = false; const micBtn = document.getElementById('micBtn');
if ('webkitSpeechRecognition' in window || 'SpeechRecognition' in window) {
  const SpeechRecognition = window.SpeechRecognition || window.webkitSpeechRecognition;
  recognition = new SpeechRecognition(); recognition.lang = 'zh-TW'; recognition.interimResults = false;
  recognition.onresult = function(event) {
    if (currentMode !== 'voice') return;
    const resultText = event.results[0][0].transcript;
    resultDiv.innerHTML = "聽到的語音:\\" + resultText + "\\";

    // 語音指令：最後一個參數傳 "1"，代表啟動 2 秒定時
    if (resultText.includes("前進") || resultText.includes("向前") || resultText.includes("直走")) {
      statusDiv.innerHTML = "語音觸發：[前進] (2秒後自動停)"; sendMoveCommand("FORWARD", "1");
    } else if (resultText.includes("倒車") || resultText.includes("後退") || resultText.includes("退後")) {
      statusDiv.innerHTML = "語音觸發：[後退] (2秒後自動停)"; sendMoveCommand("BACKWARD", "1");
    } else if (resultText.includes("左轉") || resultText.includes("向左")) {
      statusDiv.innerHTML = "語音觸發：[左轉] (2秒後自動停)"; sendMoveCommand("LEFT", "1");
    } else if (resultText.includes("右轉") || resultText.includes("向右")) {
      statusDiv.innerHTML = "語音觸發：[右轉] (2秒後自動停)"; sendMoveCommand("RIGHT", "1");
    } else if (resultText.includes("停下") || resultText.includes("煞車")) {
      statusDiv.innerHTML = "語音觸發：[停止]"; sendMoveCommand("STOP", "0");
    }
  }
}
```

學生學習成效分析

本專題適合於國高中科技領域專題課程實施，能有效引導學生打破過往對微控制器「一對一」的僵化指令，建構全新之 AIoT 跨領域系統整合觀念。在人機互動與 AI 技術體驗層面，透過導入 LLM 之語意辨識機制，讓載具得以精準解析「前進」、「向前」甚至「出發」等多元自然口語，此舉不僅能讓學生切身體會自然語言介面 (natural language inference, NLI) 之語意容錯性與靈活性，亦大幅增強學生在專題實作之成就感與主動探索之動機。

而在運算思維與工程問題解決層面，學生在面對語音訊號傳輸與機電反應之實務情境中，能展現出高度之邏輯分析與除錯能力；例如當學生觀察到語音指令發送後小車會持續行駛之現象時，能主動發起小組討論並進程式修正進行調整，展現由問題發現到流程優化之工程思維脈絡。綜上所述，本專題成功將中學階段科技領域加以統合，並有效培育國高中學生具備因應未來科技所需之跨領域系統整合實務與人機互動 (human-computer interaction, HCI) 設計之核心素養。

教學實施與注意事項

在教學場域之部署與通訊層面，本系統基於區域網路 (local area network, LAN) 架構進行資料傳輸。因此，現場教學環境之無線訊號傳輸品質與頻寬穩定度為影響人機互動延遲性之關鍵因素，使用者操作之終端控制介面（如筆記型電腦）與 ESP32 邊緣微控制器須連結至同一無線網路或行動熱點，方能順暢完成語音封包之解析與語音指令之即時回傳。此外，因 ESP32 開啟 Wi-Fi 模組進行持續性高強度

資料傳輸時會產生較高之峰值電流需求，供電系統須配置具備高穩定電壓輸出之鋰電池（如 18650 鋰電池），以避免微控制器因電壓突降而導致系統重啟或網路中斷，影響載具行駛的流暢度與穩定度。

而在硬體機電整合層面，傳統塑膠齒輪箱之直流減速馬達（如黃色 TT 馬達）受限於材料剛性與製造公差，其齒輪嚙合誤差常導致兩側輸出軸產生轉速不對稱現象，使載具在直走行駛時產生非預期的偏差。為優化載具之行駛穩定度，建議於實務範例中改採金屬齒輪減速馬達，或於嵌入式程式端導入脈衝寬度調變（PWM）速度補償機制，針對左右側馬達輸出進行微調校正，以維持車體直線運動之精準度。

未來展望

本專題將持續秉持「低硬體門檻、高 AI 體驗」之教學核心理念，從軟體普及、感測整合與多模態互動三大教學面向進行課程深度與廣度的拓展，為國、高中科技領域提供更具前瞻性與易推廣性的 AIoT 跨領域教學教材：

一、降低學校設備門檻，實踐 AI 教育普及化

本專題最大之教學突破，在於學生無需另外購買市面上昂貴且功能封閉之專業語音辨識晶片。教學上僅需利用課堂現有的電腦設備與低成本 ESP32 主控板，結合 Web API 與雲端 LLM，即可讓學生在極低的硬體預算下，體驗最主流的自然語言控制。未來將進一步模組化此教學流程，協助更多基層學校克服科技教育設備昂貴的痛點，落實 AI 普及化的教育理念，讓每位學生都能輕鬆邁入人工智慧科技的大門。

二、融入多元感測元件，深化工程問題解決與運算思維

在後續的進階課程規劃中，本專題的載具可作為彈性極高的教學平台。未來將引導學生於小車上組裝超音波感測器、紅外線循跡模組或光敏元件等硬體。透過將「AI 雲端語音指令」與「邊緣感測器即時數據」的結合，學生已不再只是設計單一遙控車，而是能符應「環境感知與自動避障」等高階情境，藉此培育學生軟硬體整合以解決真實世界問題的工程思維。

三、整合手勢辨識與機構設計，實現多模態人機互動

為提升專題實作之跨領域深度，本教案規劃在載具上加裝抬升機械臂和可開合夾爪，並導入 MediaPipe 手勢辨識技術，讓學生在未來能同時運用「語音指令」與「手勢動作」來精準控制小車行駛、手臂抬升與夾取物件。透過將語音與手勢結合的人機

互動專題，能激發學生對智慧製造與機器人開發的興趣，引導其從單純的載具操控，進階成為具備前瞻科技整合能力的 AI 創新者。

參考文獻

教育部（2018）。十二年國民基本教育課程綱要國民中小學暨普通型高級中等學校—科技領域。國家教育研究院。

陳侑成（2022）。AIoT 結合智慧製造技術發展現況與展望。機械工業雜誌，466，23-29。

附件

❖ 本教材官網網址 <https://reurl.cc/p8pqz4>

