

行政院國家科學委員會專題研究計畫成果報告

智慧型車輛之即時監控系統設計

計畫編號：NSC90-2213-E009-101

執行期限：90年8月1日至91年7月31日

主持人：徐保羅 國立交通大學電機與控制工程系

摘要

本計畫成果以三個部分來呈現，分別為(1)智慧型車輛上的自動控速系統模擬器(automatic cruise control simulator)的設計與實現，(2)車輛用感測器與車輛用網路 control area network (CAN) 之整合研究，(3) Linux 作業系統與 CAN Bus 模組的結合應用。

對於智慧型車輛上的自動控速系統模擬器設計與實現，本部份提出一個以物件導向(object-oriented)為基礎的設計方式，利用統一建模語言(unified modeling language, UML)來建造智慧型車輛上自動控速系統模擬器的模型，並進一步以爪哇語言 (Java) 來實現。所實現之模擬器，將有利於未來車輛控制系統的模擬。

在車輛用感測器與 CAN Bus 之整合研究方面，我們將在車上裝置有許多感測器，透過 CAN 來傳輸來自個別的感測器所擷取的資料，進一步實現以 CAN 為基礎的高階協定 CANopen，並以 Linux 為作業系統。

關鍵字：智慧型車輛、自動控速系統模擬器、車用網路、control area network (CAN)、CANopen、Linux、Java。

Abstract

This report includes three parts (1) the automatic cruise control simulator, (2) CAN bus for vehicles, and (3) Linux operating systems. We propose the UML modeling technique and realize it with the Java language. Furthermore, we applied the CANopen for the Linux operating

systems to achieve the network-based control for vehicles.

Keywords: intelligent vehicle, CAN, CANopen, Linux..

一、前言

在智慧型車輛中，自動控速系統 (automatic cruise control system) 是其中重要的一環。許多自動控速的方法已被提出，來控制車輛油門與剎車等實體機構，而達到要求的速度[1-5]。然而，若以實際車輛來驗證各種自動控速方法中，不但耗時、危險，也增加成本。因此，對於智慧型車輛上的自動控速系統設計，本部分的主要目的是設計與實現一車輛模擬器 (simulator)，藉由此模擬器來驗證與評估各自動控速的方法。模擬器是以統一建模語言 UML [6] 來設計，並用 Java 語言來實現 [7-8]。

在車輛用感測器與 CAN Bus 之整合研究方面，由於科技的發達，使得在車子上具有很多的 sensors，例如溫度感測器、GPS、ABS 模組及測量兩車間的超音波與紅外線模組等 [9, 10]，如何整合這些模組，必須透過車內的 CAN 來對每一個 sensor 做資料的存取，藉由這些資料來控制速度控制器模組，避免發生車輛間的碰撞或其他不安全的狀況等。在車內的網路，是透過 CAN Bus 來做佈局，本計劃將利用標準的高階協定 CANopen，與的感測器與及人機介面做一個合併，藉以達到在車上一個完整系統的整合。

在 Linux 作業系統與 CAN Bus 模組的結合應用方面，CAN 是在 1990 年由德國 Robert

Bosch 公司所訂定的一個網路協定。當初訂此協定的目的是在以一种新的數位網路技術，取代原先在汽車內複雜且昂貴的硬體接線。由於它提供了快速且可靠的連線反應，適用於即時系統上，並且費用低廉，故也被用於許多其他的控制場合。CAN 並已成為國際標準規格(ISO 11898, ISO 11519)。本計劃預計將 Linux 作業系統也加上 CAN 模組，進而能和輪椅車內的 DSP 控制器(內含 CAN 模組)或其他有 CAN 模組的感應器或為控制器作資料交換，再透過 Linux 上 HTTP Server 將資料傳到 Internet 上，以達到遠端監控和控制的目的。

接下來，我們將本計畫初步的研究成果將以三個部分來呈現，(1)自動控速系統模擬器設計與實現，(2) 在車輛用感測器與 CAN Bus 之整合研究，(3) Linux 作業系統與 CAN Bus 模組的結合應用。

二、自動控速系統模擬器設計與實現

本部分研究首先是以物件導向為基礎，來建造自動控速系統的模型。我們提出的自動控速系統模擬器之設計與實現流程如 Fig. 1 所示。

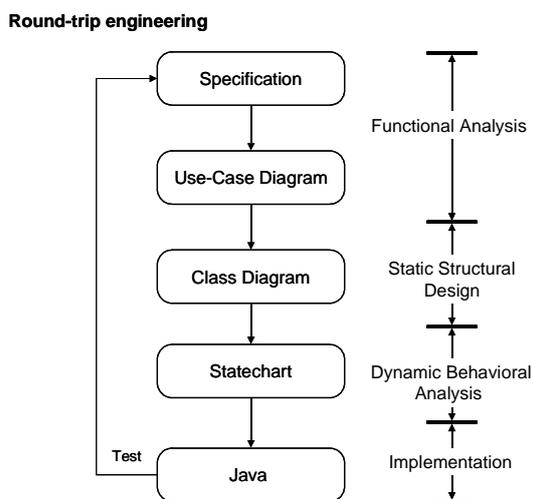


Fig. 1. 自動控速系統模擬器之設計與實現流程

為有利於之後監控系統的模擬，我們以標準的統一建模語言(unified modeling language, UML) 與 Java 為基礎 [6-8]。並初步應用於本研究中，實現車輛自動控速系統的模擬器，如

Fig. 2-4 所示。其中 Fig. 2 是系統之使用案例圖 (use-case diagram)，用於表達自動控速系統中的基本要求功能。Fig. 3 是系統之類別圖(class diagram)，用於表達系統物件之間的靜態之聯結關係。Fig. 4 是系統之狀態圖(statechart)，用來分析系統之動態行為。而 Fig. 5 是利用 Java 來實現之自動控速系統之模擬器。

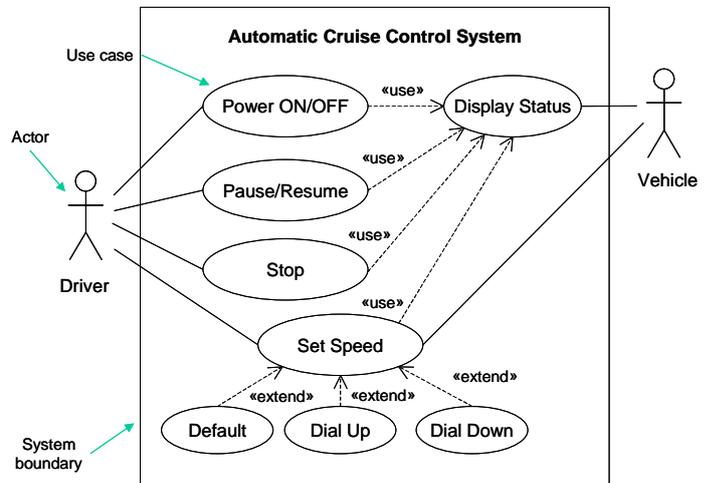


Fig. 2. 控速模擬器之功能分析：使用案例圖

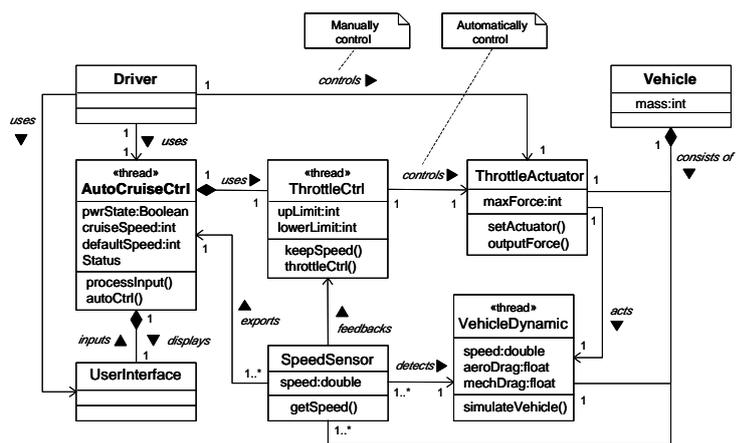


Fig. 3. 控速模擬器之靜態分析：類別圖

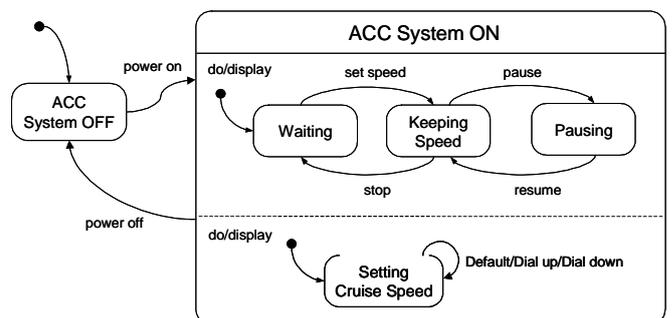


Fig. 4. 控速模擬器之動態分析：狀態圖

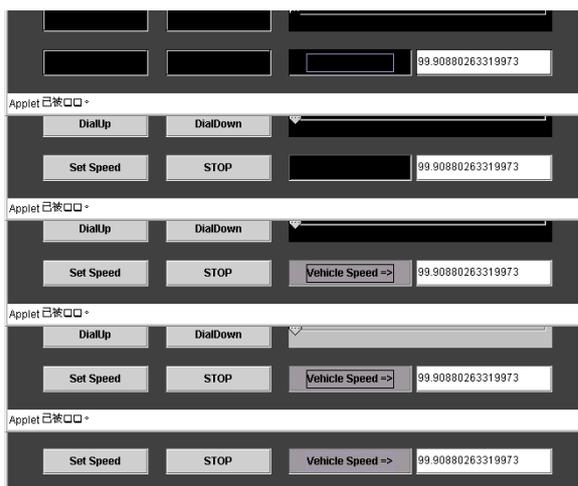


Fig. 5. 自動控速系統之 Java 模擬器

三、車輛用感測器與 CAN Bus 之整合研究

隨著科技的進步，人們對於生活的舒適性與安全性要求也相對增加。在智慧型車輛上，安全性特別是所著墨甚多的一大主題，如何增加駕駛者在駕駛上的安全性或者是藉由某些感測器來加以輔助駕駛者，一直是努力去達到的目標。在此，本計劃利用超音波與紅外線來達到兩車間距離的有效監控，考慮到超音波模組的發射與接收角度，以及架構了多個超音波模組之後，為了避免感測器間彼此互相干擾，減少測量誤差及誤動作等發生，有關於超音波模組的配置方式是必須進一步探討的。

首先考慮到車上多個感測器的存在，如 Fig. 6，因此必須有一個中控系統來對每一個 sensor 所回傳的資料做存取。在車內網路的佈局上選擇了以 CAN Bus 來作為中控系統到各個 sensor 間的資料連接，在每一個 sensor 配置一個單晶片，經由 CAN bus 可以將資料傳回給中控系統 (Notebook)，在中控系統上可以顯示出一些駕駛者需要的資訊，例如目前的速度、與前車間的距離或者是地圖資訊等。在經過資料處理後，中控系統來決定該發生什麼訊息給駕駛者或者直接控制車子的速度控制器模組。

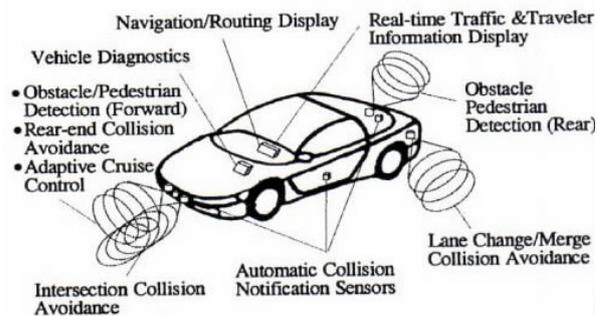


Fig. 6. 車輛用的感測系統

四、Linux 作業系統與 CAN Bus 模組的結合應用

本計劃預計將 Linux 作業系統也加上 CAN 模組，進而能和輪椅車內的 DSP 控制器(內含 CAN 模組)或其他有 CAN 模組的感測器或為控制器作資料交換，再透過 Linux 上 HTTP Server 將資料傳到 Internet 上，以達到遠端監控和控制的目的。其硬體架構如 Fig. 7 所示。

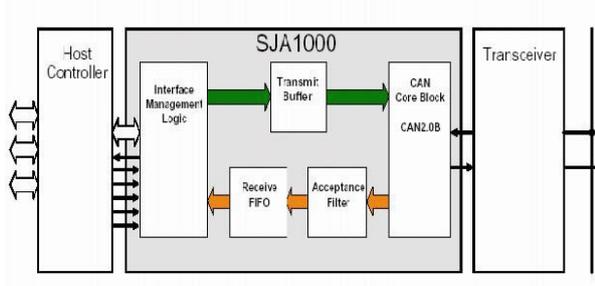


Fig. 7. Linux 與 CAN 結合之硬體架構

利用 Linux 的 Printer Port 和 PHILIPS CAN Controller SJA1000 結合，SJA1000 再和 PCA82C251 準位轉換器連接到 CAN Bus 線上，接收或傳送訊號。實驗硬體如 Fig. 8 所示。



Fig. 8. Linux 與 CAN 結合之實驗硬體

實驗 Host 端的接收與傳送畫面如 Fig. 9 所示，左端為接收的資料，包含 ID number、Data[0]~Data[7]與傳送時間。本研究以 HTTP Server 為基礎之監控架構，目前已經完成可從遠端 PC 透過瀏覽器，讀取實驗車上 CAN Bus 的資料。

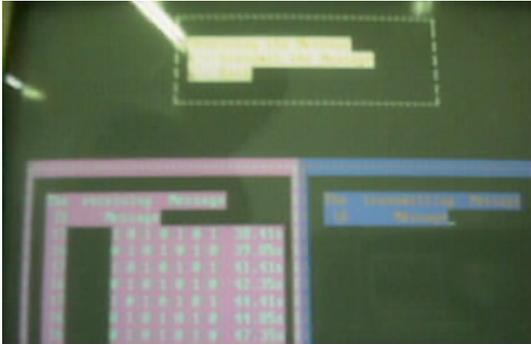


Fig. 9. Host 端的接收與傳送畫面

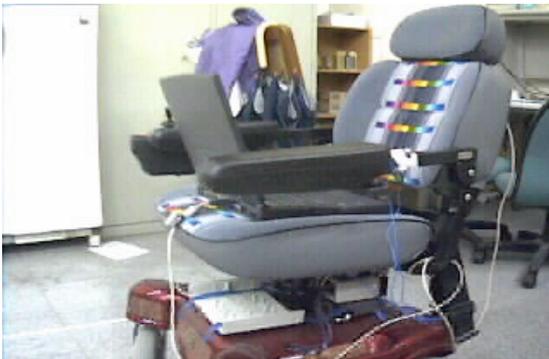


Fig. 10. 輪椅車實驗平台

五、計畫成果自評

- 研究計畫與原計畫相符程度為 85% 對於如何設計 CAN, Linux, 及 CANopen, 尚待進一步整合研究.
- 達成車輛之混合式系統模擬介面, 以 Java 為介面之模擬系統.
- 本研究結果未來將進一步實現在電動車上.

參考文獻

- [1] Y. Zhang, E. B. Kosmatopoulos, P. A. Ioannou, and C. C. Chien, "Autonomous

intelligent cruise control using front and back information for tight vehicle following maneuvers," IEEE Trans. Veh. Tech., vol. 48, no. 1, pp. 319-328, 1999.

- [2] P. Li, L. Alvarez, and R. Horowitz, "AHS safe control laws for platoon leaders," IEEE Trans. Contr. Syst. Tech., vol. 5, no. 6, pp. 614-628, 1997.
- [3] D. N. Godbole and J. Lygeros, "Longitudinal control of the lead car of a platoon," IEEE Trans. Veh. Tech., vol. 43, no. 4, pp. 1125-1135, 1994.
- [4] P. A. Ioannou, and C. C. Chien, "Autonomous intelligent cruise control," IEEE Trans. Veh. Tech., vol. 42, no. 4, pp. 657-672, 1993.
- [5] S. E. Shladover, "Longitudinal control of automotive vehicles in close-formation platoons," ASME J. Dyn. Syst., Meas., Contr., vol. 113, pp. 231-241, 1991.
- [6] G. Booch, J. Rumbaugh, and I. Jacobson, The Unified Modeling Language User Guide. Reading, MA: Addison-Wesley, 1999.
- [7] Sun Microsystems, The Java Tutorial. <http://java.sun.com/docs/books/tutorial/>, December, 2001.
- [8] J. Greenfield, "Unified Modeling Language/Enterprise JavaBeans (UML/EJB) Mapping Specification," Rational Software Corporation Document, May, 2001.
- [9] <http://www.stakes.fi/tidecong/673schi.html>, Sensors to Improve the Safety for Wheelchair Users.
- [10] D. Hohman, T. Murdock, E. Westerfield, T. Hattox, and T. Kusterer, 2000 IEEE GPS Roadside Integrated Precision Positioning System.