

以 Arduino 微控器為基礎的行車安全警示系統 與 APP 程式設計

張添炮¹、劉芳嬌²

¹ 南開科技大學 多媒體動畫應用系

² 南開科技大學 電機與資訊技術系

通訊作者：張添炮

聯絡地址：542 南投縣草屯鎮中正路 568 號

電子郵件：t118@nkut.edu.tw

投稿日期：2017 年 5 月

接受日期：2017 年 6 月

摘 要

隨著社會的進步，現在的汽車設計愈來愈重視各種安全概念。這其中，安裝一套可自動偵測其他車輛動態的配備，就是很重要的一個項目。如果把偵測到的訊息透過無線網路傳輸到行動裝置，另做適當的運用，更能提升行車的安全係數。Arduino 是一種開放程式碼的微控器，因為具備電路簡單、功能強等優點，所以常被業界應用在專案設計中。本研究以 Arduino UNO 微控器為開發平台，嘗試以簡易物體模擬汽車的運動，搭配超音波測距模組(HC-SR04)及藍牙模組(HC-05)，將偵測到的距離值換算為相對速度和加速度。這構想可以運用在汽車的防撞偵測方面，在行車當中遇到四種預先設定的危險狀況時，系統會發出警示聲響和燈號，提醒駕駛人。另以 APP Inventor 2 為開發平台設計 Android APP 程式，把兩物體的距離、速度和加速度等訊息顯示在行動裝置上。

關鍵詞：Arduino UNO、微控器、超音波模組、藍牙模組、APP

壹、緒論

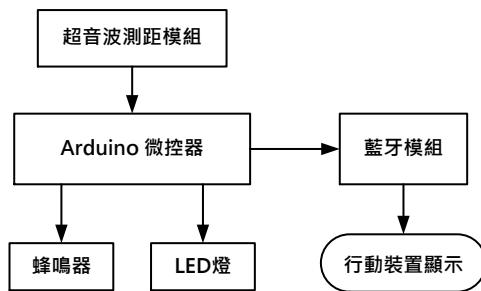
隨著經濟的發展，社會觀念的進步，汽車加裝感測設備、提升安全係數，已經普遍被民眾所接受。例如安全氣囊(SRS)、防鎖死剎車系統(ABS)、倒車輔助、定速行駛、行車紀錄器、自動停剎、後方來車警示等等。這其中有一些項目是利用車子與物體的相對距離做為判斷的依據。有關於距離的偵測，超音波或者雷達波是較常用的兩種媒介。

Arduino 是一種開放程式碼(Open source)的微控器(Microcontroller)，不僅軟體，就連硬體也是開放的(請參考 Arduino 官方網站、維基百科 Arduino、位明先，2015、艾迪

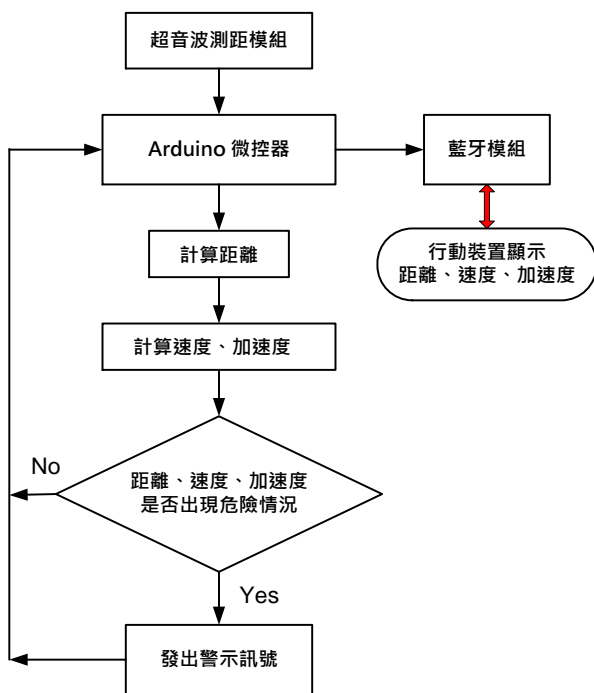
諾，2016、Arduino.TW 台灣使用者社)。因為具備電路簡單、功能強、體積小、耗電低、價格便宜等優點，所以近年來常被應用在電子、資訊、機械、汽車、醫療等專案設計中(請參考相關文獻：余瀧濱，2011、洪政義，2010、陳民峻，2014、賴宜廷，2014、高誌謙，2012、陳泳杉，2016、陳聖翰，2013、張文璋，2011、沈之中，2011)。與行車安全相關的著作則請參考下列文獻：林秉威等(2017)、徐理芳等(2015)、許文豪等(2016)、郭芳兩與鐘毓驥(2016)、陳冠廷等(2015)。微控器如同一台小型電腦，負責命令各電子元件的動作，可以執行豐富多樣化的無線網路專案。

本研究嘗試以簡易物體模擬汽車的運動，利用 Arduino

微控制器，搭配超音波測距模組 HC-SR04(請參考相關網站: Arduino 超音波模組、超音波測距模組、HC-SR04 超音波模組、Matlab HC-SR04 庫存函數)，根據音速公式算出兩物體的距離、相對速度、加速度。本研究另以 APP Inventor 2 為開發平台撰寫 APP 程式(參考 APP Inventor 2 官方網站、文淵閣工作室)，並透過藍牙模組 HC-05 (參考網站: 藍牙模組介紹、HC 藍牙系列、HC-05 藍牙模組)，把距離、速度和加速度等訊息顯示在 Android 手機螢幕上。我們根據距離、速度和加速度的大小，事先設定四種危險狀況。在行車過程當中遇到任何一種危險狀況時，系統會發出警示聲響和燈號，提醒駕駛人。系統架構如圖一所示，軟體運作流程如圖二所示。



圖一 系統架構圖



圖二 軟體運作流程圖

貳、Arduino 微控制器

本文所使用的 Arduino 微控制器型號是 UNO R3 (圖三)，它是用 ATmega328P 型號的 CPU，有 32KB 的撰寫程式空間(Flash ROM)與 2KB 的暫存空間(RAM)。提供 14 個數位

輸入/輸出腳位(D0~D13)，其中第 0 和 1 腳位分別用來接收(RX)和傳輸(TX)TTL 訊號的序列資料。而第 3、5、6、9、10、11 等六個腳位，則可以提供 8-bit 的 PWM(Pulse Width Modulation)輸出。第 13 腳位可以控制內建的 LED。另有 6 個類比輸入腳位(A0~A5)，提供 10 位元的解析度，即 1024 種不同的數值。微控制器所需工作電力可由電腦的 USB 或是外部直流電源供應。Arduino 支援多種電子周邊模組，例如感測器、控制器、發光二極體、步進馬達或其他輸出裝置。其控制程式可在 Arduino IDE 整合開發環境 (Arduino Integrated Development Environment)下撰寫(參考網站: Arduino IDE、Cooper Maa 筆記)，再編譯、上傳到微控制器。程式語法類似 Java、C 語言。Arduino 晶片可以重複燒錄程式。Arduino IDE 為開放原始碼，可讓使用者免費下載安裝。



圖三 Arduino UNO R3 微控制器

參、超音波測距模組

聲波是一種機械波，必須透過介質才能夠傳遞。聲波由物體振動產生，人耳可以聽到的聲波頻率約在 20~20000Hz 之間。超過 20000Hz 就是超音波(Ultrasound)。超音波因為頻率高、波長短，所以指向性特別好，能量消耗緩慢，在介質中傳播的距離較遠，常用來偵測物體的距離，例如測距儀、胎兒診斷、魚群追蹤、測速照相、探勘考古等。聲波在空氣中的傳播速度，可用下列公式表示：

$$v \text{ (cm/s)} = 33150 + 60 t \quad (1)$$

其中 t 為攝氏溫度(C)。本研究使用的超音波測距模組(HC-SR04)如圖四所示，是由超音波發射器、接收器和控制電路所組成，有效測量範圍在 2~450 公分。超音波傳遞遇到障礙物時會反射，可藉由發射與接收的時間差，透過音速公式，求得障礙物的距離。而速度相當於每單位時間內距離的變化量；加速度則相當於每單位時間內速度的變化量。HC-SR04 常應用在自走車、物體近接、汽車防碰撞偵測等方面的設計。HC-SR04 共有 Vcc、Trig、Echo 與 GND 四個腳位。

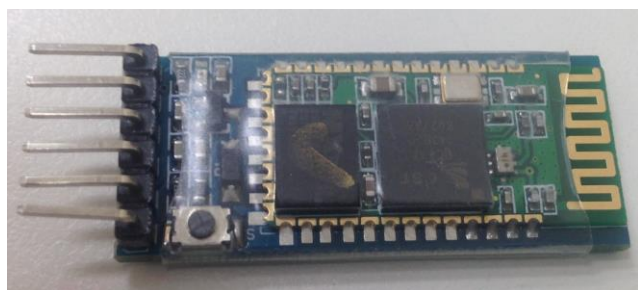
超音波的發射與接收，是靠 Trig 與 Echo 這兩個腳位，Trig 為輸入，Echo 為輸出。Vcc 與 GND 分別接 5V 電源及 GND。Trig 接第 8、Echo 接第 9 數位腳位。先拉低 Trig 電位，再送出 10 微秒的 5V 高電位訊號給 Trig，模組會自動發射 8 個 40KHz 的超音波(方波)，接著 Echo 腳位就會處於 5V 高電位狀態。如果收到反射訊號，那麼 Echo 腳位就會處於 0V 低電位狀態。所以 Echo 處在高電位的時間即是超音波從發射到接收的時間。



圖四 HC-SR04 超音波測距模組

肆、藍牙模組

本研究使用的 HC-05 模組是主從一體的藍牙序列模組，如圖五所示。可以在 AT Command 環境下設定模組的特性參數，例如，藍牙名稱、配對密碼、主從模式、位址、波特率等。HC-05 使用時必須先進行搜尋，當主從兩個藍牙模組配對成功後，就可以取代傳統的序列線通訊，改用無線的藍牙方式。雖然使用方式較為複雜，但能夠利用程式設定各種不同的通訊參數，故可用於各種帶藍牙功能的電腦、手機、平板等智慧型裝置。HC-05 適用電壓範圍為 3.6~6V，波特率支援範圍是 4800~1382400 bps，在無遮蔽時，有效連線距離約 10m。其接線方式：Vcc 與 GND 分別接 5V 電源及 GND，Tx 接第 11、Rx 接第 10 數位腳位。



圖五 HC-05 藍牙模組

伍、成果討論

一、Arduino 程式設計

控制程式是在 Arduino IDE 整合開發環境之下撰寫，如

附錄一所示。除了超音波和藍牙模組的宣告，本研究也加入蜂鳴器和 LED 燈兩種警示元件。蜂鳴器接第 12、LED 燈接第 13 數位腳位。圖六所示為兩物體(汽車)相對運動示意圖，超音波模組可以安裝在汽車前後保險桿位置。首先，利用 Echo 腳位高電位持續時間與音速公式求出物體的距離。假設氣溫為攝氏 25 度，其距離公式為：

$$d \text{ (cm)} = (T/2) * 0.03465 \tag{2}$$

其中 T 為高電位持續時間。根據系統內部的時間變化，換算出兩物體的相對速度和加速度。再將兩物體的距離、速度、加速度等資料傳送到手機。值得注意的是，為了確保資料能夠準確傳送，Arduino 在發送訊息之前，會先檢查是否收到手機端送來的識別碼(Key，本研究設定為 118)，當確認無誤之後，就會將訊息送出去。而在資料的最前面也會附上 Arduino 傳給手機端的識別碼，讓手機知道可以開始接收資料了，避免發生封包漏接的情況。手機端每次接收資料，都會以識別碼為基準開始接收，除了識別碼之外，另外再接收六個 Bytes(1 Byte=8 bits)的資料。本研究為了保留小數兩位，三種資料都先乘上 100，並化成整數再轉成兩個 Bytes 的封包傳送。若物體在運動過程中，出現表一所示的四種危險情況，系統就會發出警示訊號；蜂鳴器發出聲響，LED 燈亮起。第 1 種情況是，兩物體(汽車)距離在 15cm 以內，不論速度、加速度大小為何。第 2 種情況是，兩物體距離介於 15~60cm 之間，而且速度大於 6cm/s。第 3 種是，兩物體距離大於 60cm，而且速度大於 10cm/s。第 4 種是，兩物體相對加速度大於 100cm/s²，不論距離、速度大小為何。



圖六 兩物體(汽車)相對運動示意圖

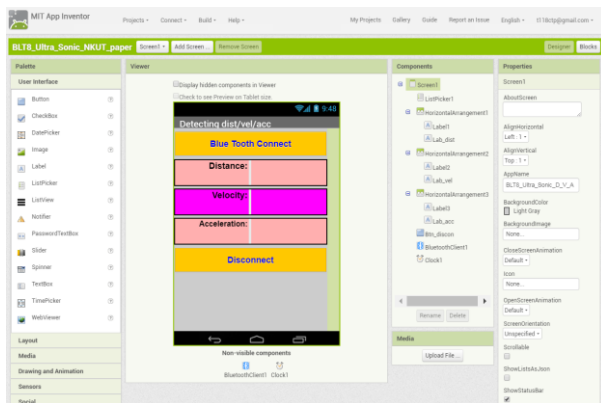
表一 本研究所考慮的危險狀況

#	condition
1	$d \leq 15$
2	$(15 < d \leq 60) \ \& \ (v \geq 6)$
3	$(d > 60) \ \& \ (v \geq 10)$
4	$a > 100$

d: distance(cm), v: velocity(cm/s), a: acceleration (cm/s²)

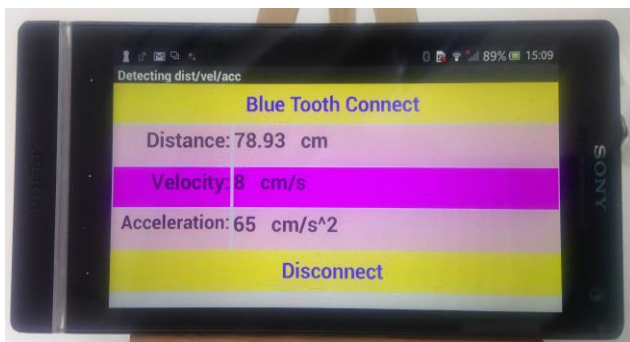
二、App 程式設計

本研究使用 MIT App Inventor 2 這個免付費的雲端創作平台，進行 APP 程式設計[.]。因為它使用拼裝圖塊的方式來組合程式，又提供各式各樣的元件，所以簡單易學、方便實用[文淵]。用這個平台所開發的 APP 程式適用在 Android 系統裝置。App Inventor 2 使用者介面如圖七所示，這其中有本研究所設計的 APP 畫面。APP 程式碼如附錄二所示。



圖七 本研究所設計的 APP 畫面

當藍牙配對成功以後，每隔一段時間，手機會去接收來自藍牙的訊息。要注意的是，為了避免封包漏接的情形，計時器元件(Clock)的時間間隔(Time Interval)不可以定得太長。在實務應用上，時間間隔愈短愈能反映行車的即時動態。本研究嘗試以 100ms 為例作說明。計時器啟動之後，每隔 100ms 就會動作一次，手機先發送一個識別碼(=118)給 Arduino，Arduino 確認無誤之後，也會發送一個識別碼(=118)給手機。經雙方確認無誤後，手機開始接收六個 Bytes 的數值，因為 Arduino 送出的數值是採用 8bits 的封包，所以必須將它從 8bits($2^8 = 256$)轉回 10 進制，最後再除以 100 轉成正確的單位(cm、cm/s、cm/s²)。如圖八，手機螢幕顯示兩物體相對運動的即時數據。



圖八 兩物體相對運動的即時數據顯示在手機螢幕

陸、結論

本研究利用簡易的物體模擬道路上行駛的汽車，物體可能處在靜止或運動狀態，進行多次實驗。首先用超音波模組，配合音速公式，求出兩物體的相對距離；再根據系統內部的時間，換算出相對速度和加速度。若出現預設的危險行車狀況，系統就會發出警示訊號。另一方面，本研究設計一套 APP 程式，透過藍牙系統，把資料顯示在 Android 手機螢幕。結論可歸納如下：

- 一、以 Arduino UNO 微控制器為開發平台，利用超音波模組測量兩物體的距離，進一步求出相對速度和加速度的做法，是本研究的創見之一。
- 二、雖然 HC-SR04 超音波模組的有效測距範圍不夠遠，但本文所提出來的概念，仍可以被延伸應用在行車安全設計方面。把測距模組安裝在汽車的前端或者後端，當出現預設的危險狀況時，Arduino 系統可以發出警報提醒駕駛人。未來如果採用更精進的測距裝置，將可提升它的實用性。
- 三、把兩物體的相對距離、速度、加速度等訊息，透過藍牙顯示在手機螢幕，是本研究的創見之二。未來可以做更深一層的運用，提升行車安全係數。如果汽車本身具備藍牙影音系統，例如 Android Auto 或 Carplay 系列，則可以顯示在汽車系統。
- 四、汽車本身的行駛速度並未在本研究中列入考慮，這是未來值得研究的方向。
- 五、在使用上，因為超音波模組的運作速度非常快，可能導致藍牙手機無法正確的顯示訊號。因此 Arduino 驅動超音波模組的時間間隔要和手機 APP 接收訊息的時間互相配合，這是未來研發設計上所必須注意的一個細節。

參考文獻

- 文淵閣工作室，<http://www.e-happy.com.tw/>。
- 艾迪諾 (2016)。Arduino 全能微處理機實習。台北市，全華圖書公司。
- 余瀧濱(2011)。於 Android 智慧型手機以近場通訊啟用藍芽資料傳輸之研究。國立臺灣師範大學資訊工程研究所碩士論文，台北市。
- 位明先 (2015)。Arduino 微電腦專題製作。台北市，台科大圖書公司。
- 沈之中(2011)。基於 Android 平台實現即時藍牙無線心電訊號感測與處理系統。國立中正大學通訊工程研究所碩士論文，嘉義縣。
- 林秉威、黃思碩、陳松雄(2017)。兩車聯行循跡自走車製作。修平科技大學資訊網路技術系實務專題論文，台中市。
- 洪政義(2010)。藍芽結合 NFC 於門禁通行控管之安全評量。

- 大同大學資訊工程系碩士論文，台北市。
- 徐理芳、林亞宏、顏金泰(2015)。遙控車擴充應用-以APP軟體設計。中華大學資訊工程學系專題論文，新竹市。
- 高誌謙(2012)。基於Android與ZigBee之智慧家庭設計與實現。國立高雄海洋科技大學電訊工程研究所碩士論文，高雄市。
- 張文璋(2011)。智慧藥盒結合Android平台之藍芽行動看護系統情境開發。國立臺灣大學機械工程學研究所碩士論文，台北市。
- 許文豪、池慶龍、鄭喬植、袁榮宏、吳家瑋、吳修安(2016)。駕駛狀態即時回報系統。Conference on Information Technology and Applications in Outlying Islands, PP. 281-288，高雄市。
- 郭芳雨、鐘毓驥(2016)。Arduino機車減速警示系統。長榮大學資訊工程學系專題論文，台南市。
- 陳冠廷、劉凱平、林義楠(2015)。利用Arduino對行車安全及動態之研究。明志科技大學電子工程系專題論文，新北市。
- 陳民峻(2014)。基於Arduino及影像處理的居家全監控系統。國立彰化師範大學電機工程學系碩士論文，彰化縣。
- 陳泳杉(2016)。以Arduino整合感測器實現物體偵測之應用。元智大學資訊管理學系碩士論文，桃園市。
- 陳聖翰(2013)。Android裝置控制Arduino機器人之研究。國立雲林科技大學電子與光電工程研究所碩士論文，雲林縣。
- 超音波測距模組，<http://www.micropik.com/PDF/HCSR04.pdf>。
- 維基百科Arduino，<http://zh.wikipedia.org/wiki/Arduino>。
- 賴宜廷(2014)。基於Arduino與Android之智慧家庭遠端監控系統設計與實作。國立虎尾科技大學機械與機電工程研究所碩士論文，雲林縣。
- 藍牙模組介紹，<https://swf.com.tw/?p=712>。
- APP Inventor 2官方網站，<http://appinventor.mit.edu/explore/>。
- Arduino官方網站，<https://www.arduino.cc/>。
- Arduino超音波模組，<https://playground.arduino.cc/Code/SR04>。
- Arduino IDE下載，<http://arduino.cc/en/Main/Software>。
- Arduino.TW台灣使用者社<http://arduino.tw/index.php>。
- Cooper Maa 筆記，<http://coopermaa2nd.blogspot.tw/search/label/Arduino>。
- HC藍牙系列，<http://www.wavesen.com/index.asp>。
- HC-05藍牙模組，<http://shop.cpu.com.tw/product/45456/pdf2/>。
- HC-SR04 超 音 波 模 組 ， <http://shop.cpu.com.tw/product/45466/info/>。
- Matlab HC-SR04 庫存函數，<https://www.mathworks.com/matlabcentral/fileexchange/57898-hc-sr04-add-on-library-for-arduino?focused=6504345&tab=example>。

附錄

一、Arduino 程式碼

```
#include <SoftwareSerial.h> //Arduino I2C library
const int trig = 8; // 超音波模組 Trig腳位 D8
const int echo = 9; // 超音波模組 Echo 腳位 D9
const int RX =10; //藍牙腳位 TX 接 pin 10; RX接 pin 11
const int TX=11;
int beepin=12; //蜂鳴器腳位
int ledpin=13; //LED燈腳位
const int delay_time = 100; //delay 100 ms
long dist1=0; //為了求出速度
long dist2=0;
long dx=0;
long t1=0;
long t2=0;
long dt=0;
long v1=0;
long v2=0;
long acc=0;
float duration;
float distance;
float vel;

SoftwareSerial BT(RX, TX); //set for Bluetooth Module
void setup() {
  Serial.begin(9600); //序列視窗波特率(bps)
  BT.begin(38400); //藍牙波特率
  pinMode(trig, OUTPUT);
  pinMode(echo, INPUT);
  pinMode(beepin, OUTPUT);
  pinMode(ledpin, OUTPUT);
}

void loop() {
  digitalWrite(trig, LOW);
  delayMicroseconds(2);
  digitalWrite(trig, HIGH); //trigger Ultrasonic Module持續10微秒
  delayMicroseconds(10);
  digitalWrite(trig, LOW);
  duration = pulseIn(echo, HIGH);
  distance = (duration / 2) * 0.03465; // (in cm)
```

```
delay(delay_time); // 每間隔100ms偵測一次

t1=t2;
t2=millis();
dist1=dist2;
dist2=distance;
dx=dist2-dist1;
dt=(t2-t1);
vel=-1000*(dist2-dist1)/dt; // 正負號是考慮方向性
v1=v2;
v2=vel;
acc= 1000*(v2-v1)/dt; // 正負號是考慮方向性

Serial.print(" distance(cm)= "); // 在序列視窗檢視輸出數據
Serial.print(distance);
Serial.print(" dx(cm)= ");
Serial.println(dx);
Serial.print(" dt(ms)= ");
Serial.print(dt);
Serial.print(" vel(cm/s)= ");
Serial.print(vel);
Serial.print(" acceleration (cm/s/s)= ");
Serial.println(acc);

int send_dis = (int) (dist2*100); //乘以100只是為了保留小數位數
int send_vel = (int) (vel*100);
int send_acc = (int) (acc*100);
byte packet [7]; // 共有0~6七個封包
packet[0] = 118; //key通關密碼
packet[1] = send_dis/256; //把資料轉為兩個8位元的封包傳輸
packet[2] = send_dis%256; //餘數
packet[3] = send_vel/256;
packet[4] = send_vel%256;
packet[5] = send_acc/256;
packet[6] = send_acc%256;

if(BT.available() > 0)
  if(BT.read() == 118) //確認密碼是否正確
  {
    for(int i = 0; i < 7; i++)
      BT.write(packet[i]); //開始傳輸
  }
// 在不同的行車情況下發出警報
```

```
if (dist2 <=15) {
    digitalWrite (beepin, HIGH);
    digitalWrite(ledpin, HIGH);
    Serial.print(" condition 1 ");
}
else if (dist2 >15 & dist2 <=60 & vel >= 6) {
    digitalWrite (beepin, HIGH);
    digitalWrite(ledpin, HIGH);
    Serial.print(" condition 2 ");
}
else if (dist2 >60 & vel >= 10) {
    digitalWrite (beepin, HIGH);
    digitalWrite(ledpin, HIGH);
    Serial.print(" condition 3 ");
}
else if (acc > 100) {
    digitalWrite (beepin, HIGH);
    digitalWrite(ledpin, HIGH);
    Serial.print(" condition 4 ");
}
else
{ digitalWrite(beepin, LOW);
  digitalWrite(ledpin, LOW);
}
}
```


二、APP 程式碼

```

when ListPicker1 . BeforePicking
do
    set ListPicker1 . Elements to BluetoothClient1 . AddressesAndNames

when ListPicker1 . AfterPicking
do
    if call BluetoothClient1 . Connect
        address ListPicker1 . Selection
    then
        set ListPicker1 . Enabled to false
        set Btn_discon . Enabled to true
        set Clock1 . TimerEnabled to true

when Clock1 . Timer
do
    call BluetoothClient1 . Send1ByteNumber
        number 118
    if call BluetoothClient1 . BytesAvailableToReceive > 0
    then
        if call BluetoothClient1 . ReceiveSigned1ByteNumber = 118
        then
            set Lab_dist . Text to join
                call BluetoothClient1 . ReceiveSigned1ByteNumber * 256 + call BluetoothClient1 . ReceiveSigned1ByteNumber / 100
                " cm "
            set Lab_vel . Text to join
                call BluetoothClient1 . ReceiveSigned1ByteNumber * 256 + call BluetoothClient1 . ReceiveSigned1ByteNumber / 100
                " cm/s "
            set Lab_acc . Text to join
                call BluetoothClient1 . ReceiveSigned1ByteNumber * 256 + call BluetoothClient1 . ReceiveSigned1ByteNumber / 100
                " cm/s^2 "

when Btn_discon . Click
do
    call BluetoothClient1 . Disconnect
    set ListPicker1 . Enabled to true
    set Btn_discon . Enabled to false
    set Clock1 . TimerEnabled to false
    
```

An Alert System and APP Design for Driving Safety Based on Arduino Microcontroller

Tian-Pau Chang¹, Feng-Jiao Liu²

¹Department of Multimedia Animation and Application, Nankai University of Technology, Taiwan

²Department of Electrical and Information Technology, Nankai University of Technology, Taiwan

Abstract

With the progress of the society, the security concepts concerning drive have currently been paid more and more attentions in car design. Among them, installing a set of device to automatically detect the adjacent vehicle's dynamic situations is important part. If one can transmit the detected information through wireless network to a mobile device and makes further analysis for the driver, the safety factor would be enhanced. Arduino is an open-source microcontroller; it is commonly used in the industry project for its simple and powerful circuit features. In this study, the Arduino UNO microcontroller was employed as the development platform incorporating the ultrasonic module (HC-SR04) and Bluetooth module (HC-05). We simulate the movement of a vehicle with simple objects and compute both the velocity and the acceleration of the vehicle by the detected relative distance. This concept can be applied to the collision detection of vehicle; the proposed system will alert a warning sound and light to remind the driver when the vehicle encounters any of four pre-set dangerous conditions. An Android APP was also designed using the development platform of MIT APP Inventor 2 to show the distance, velocity and acceleration data on the mobile device.

Keywords: Arduino UNO; Microcontroller; Ultrasonic module; Bluetooth module; APP