Topic 05: Introduction to Communication by Bluetooth

Bluetooth(藍芽，BT)原是十世紀統一了丹麥的國王的名字，現取其“統一”的含義，用來命名意在統一無線區域網路通訊標準的藍芽技術。藍芽技術是Erisson、IBM等5家公司在1998年聯合推出的一項無線網路技術。隨後成立的藍芽技術聯盟（SIG）來負責該技術的開發和技術協議的制定，如今全世界已有1800多家公司加盟該組織，最近微軟公司也正式加盟並成為SIG組織的領導成員之一。

Bluetooth是無線資料和語音傳輸的開放式標準，它將各種通信設備、電腦及其終端設備、各種數位資料系統、甚至家用電器採用無線方式聯接起來。它的傳輸距離為10cm～10m，如果增加功率或是加上某些外設便可達到100m的傳輸距離。它採用2.4GHz ISM頻段和調頻、跳頻技術，使用權向糾錯編碼、ARQ、TDD和基帶協議。

由於藍芽採用無線介面來代替有線電纜連接，具有很強的移植性，並且適用於多種場合，加上該技術低功耗、對人體危害小，而且應用簡單、容易實現，所以易於推廣。

藍芽技術結合了電路交換與分組交換的特點，可以進行非同步資料通信，可以支援多達3個同時進行的同步話音通道，還可以使用一個通道同時傳送非同步資料和同步話音。每個話音通道支援64kb/秒的同步話音鏈路。非同步通道可以支援一端最大速率為721kb/秒、另一端速率為57.6kb/秒的不對稱連接，也可以支援43.2kb/秒的對稱連接。

在藍芽協議的最上部是各種高層應用框架。其中較典型的有撥號網路、耳機、區域網路訪問、檔傳輸等，它們分別對應一種應用模式。各種應用程式可以透過各自對應的應用模式實現無線通信。撥號網路應用可透過模擬串列埠訪問微微網（Piconet），資料設備也可由此接入傳統的區域網路；用戶可以透過協議中的Audio（音效）層在手機和耳塞中實現音效流的無線傳輸；多台PC或筆記本電腦之間不需要任何連線，就能快速、靈活地進行檔案傳輸和共用資訊，多台設備也可由此實現同步操作。

總之，整個藍芽協定結構簡單，使用重傳機制來保證鏈路的可靠性，在基帶、鏈路管理和應用層中還可實行分級的多種安全機制，並且透過跳頻技術可以消除網路環境中來自其他無線設備的干擾。

藍牙裝置分成master(主控)與slave(從端)兩大類型。像PC與手機的BT可以探索並與其他BT周邊裝置配對(pairing)就是master。Slave則是被動等待被連結，像藍牙滑鼠、鍵盤、耳機等。一個藍牙主控裝置(Master)可同時連結最多7個主動模式的藍牙從端裝置來形成一piconet(微網)。

SIG為了確保BT設備間的互通性，定義多種規範(*Profile*，協議)，例如:

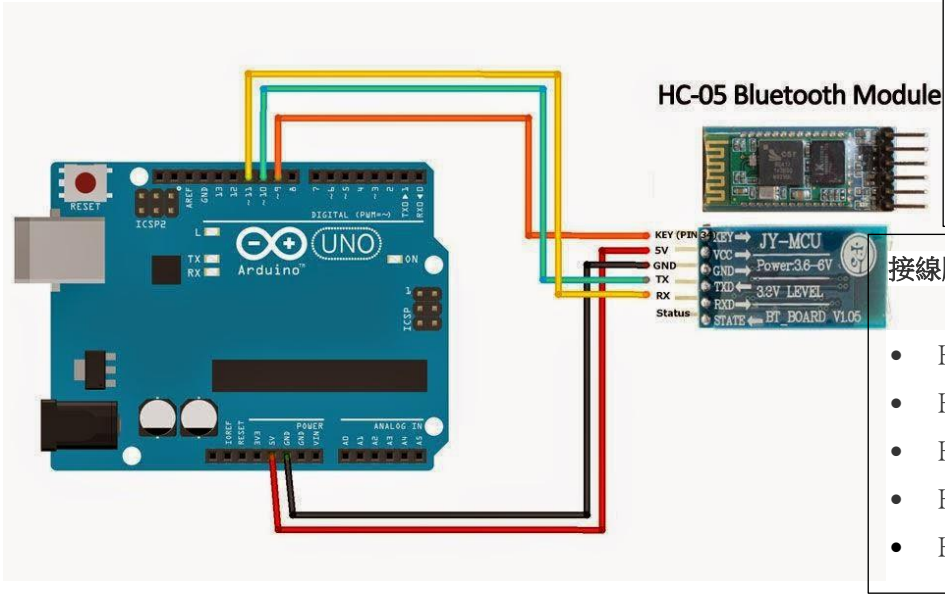
1. HID: 制定滑鼠、鍵盤與搖桿等人機介面裝置(Human Interface Device，HID)鎖鑰遵循規範。
2. HFP: 用於行動裝置以支援語音撥號和重撥等功能的免持聽筒裝置。只能傳輸8-bit、8kHz的低品質聲音。
3. A2DP: 可傳輸16-bit、44.1kHz的高品質立體聲音樂。
4. SPP(Serial Port Profile): 用於取代有線序列埠的藍芽裝置規範。

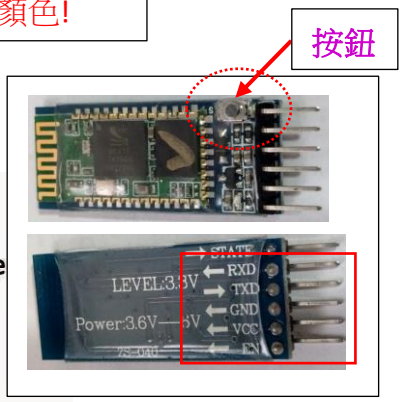
目前已有V1.1~V5.1版本。V2.0傳輸率可達576Kbps，V3.0&V4.0理論值可達24Mbps。BT V4.X和V5.0並不相容於V2.X與V3.X，像BT V4.X並不具備SPP規範。本課程將以V2.1為實驗對象。所採用的BT模組是HC-05，是一種串列埠介面的模組。該模組的工作方式是，給該模組上電之後，藍芽模組就處於一種等待被別的藍芽搜索和配對的狀態。當有別的藍芽和該藍芽模組建立配對，並建立SPP服務連接之後，該藍芽模組會自動的把它從串列埠接收到的資料發送給遠端的藍芽，同時也會把接收自遠端的藍芽的資料透過串列埠發送給相應的設備。

請注意接線，要以接腳編號為主，不是以接線顏色!

Ex5.01 Modify BT'Name by Arduino

Step.1 以杜邦線將 HC-05 與 Arduion 連接



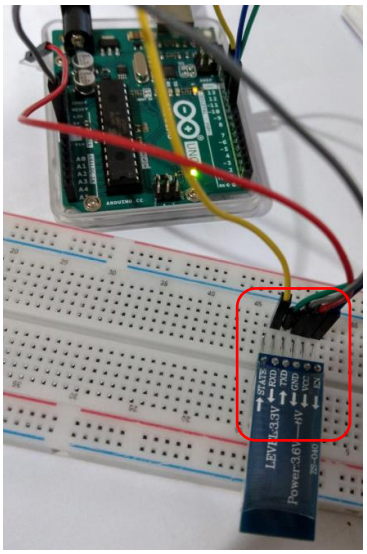


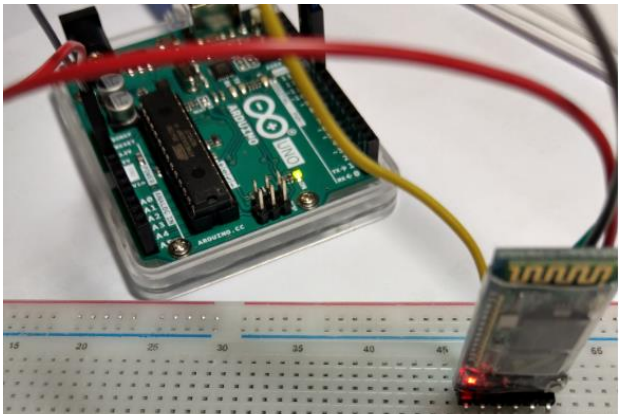
接線腳位對照表

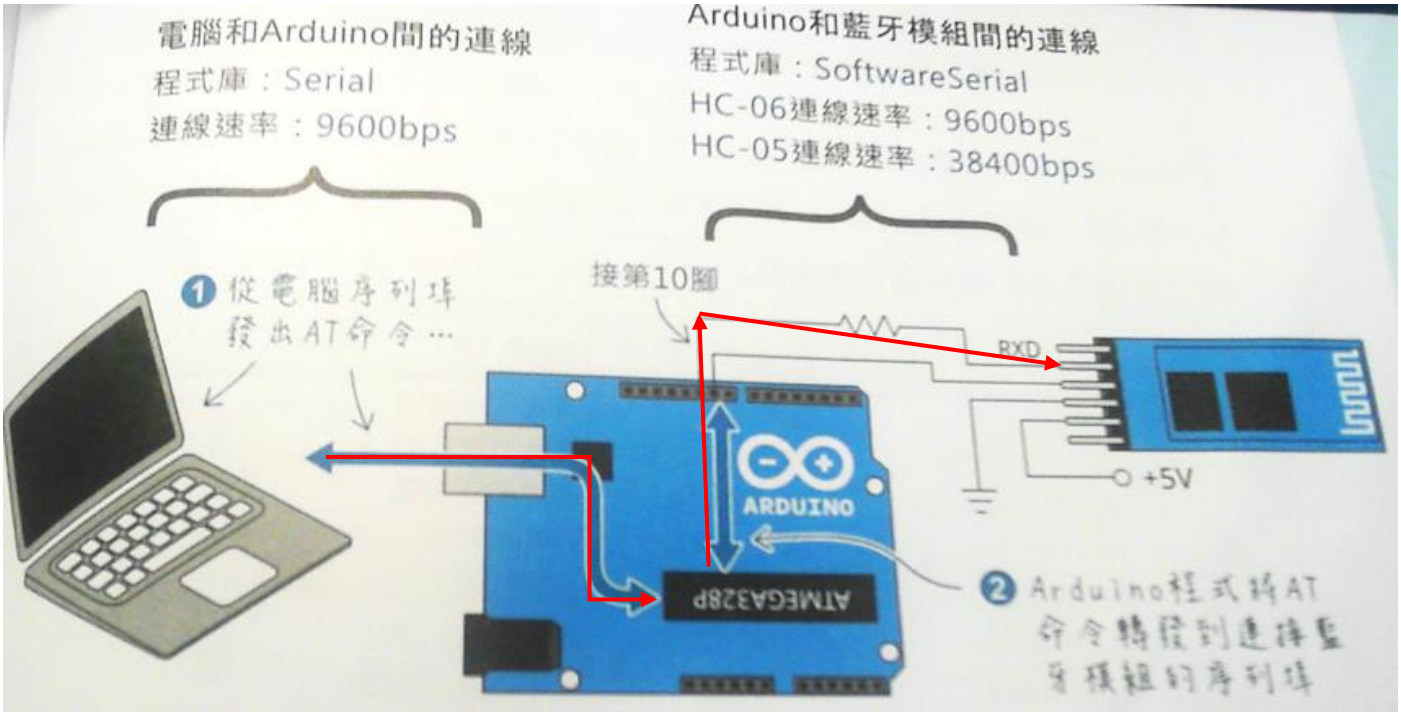
- HC-05 Key/EN <---> Arduino Pin9
- HC-05 Vcc (5V) <---> Arduino 5V
- HC-05 GND <---> Arduino GND Pin
- HC-05 TX <---> Arduino Pin10 (soft RX)
- HC-05 RX <---> Arduino Pin11 (soft TX)

Step 2 : 上傳程式碼至 Arduino (Ex5_01 Ex01_SetupBT.ino)

```
#include <SoftwareSerial.h>
SoftwareSerial BT(10, 11); //Arduino's pin-10(Receiving):BT's Tx, Arduino's pin-11(Transmitting):BT's Rx
int RESET = 9;                  // BT module uses pin 9 for reset
char val;
void setup()
{//this pin will pull the HC-05 pin 34 (key pin) HIGH to switch module to AT mode
   pinMode(RESET, OUTPUT);
   digitalWrite(RESET, HIGH);
   Serial.begin(9600);
   delay(10);
   Serial.println("BT is ready!");
   BT.begin(38400);    // HC-05 default speed in AT command
}
void loop()
{// Keep reading from HC-05 and send to Arduino Serial Monitor
   if (BT.available())
   {
      val=BT.read();
      Serial.print(val);
   }
   // Keep reading from Arduino Serial Monitor and send to HC-05
   if (Serial.available())
   {
      val=Serial.read();
      BT.print(val);
   }
}
```

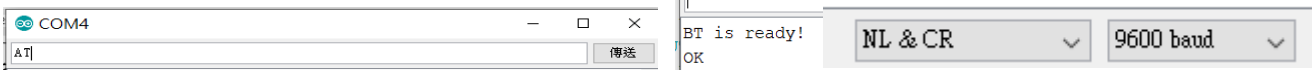






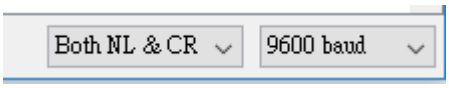
Step 3 ： 設定 HC-05 進入 AT Mode (* 注意 ： 請先將 Arduino 電源拔除再進行以下步驟)

1. 確認是否線路皆依照 Step 1 接線完成
2. 把 Arduino 接上電源前，請將 HC-05 連接至 Arudino 上的 Vcc 腳位拔除，其餘四支腳位皆維持連接狀態
3. 確認拔除 Vcc 腳位後，現在可將 Arduino 接上電源(需先按住按鈕，才可接電)
4. 確認 HC-05 目前沒有與任何藍牙裝置配對中
5. 將 HC-05 的 Vcc 腳位插回 Arduino Vcc
6. 上述步驟都正確執行後， HC-05 的 LED 燈應該維持兩秒閃爍一次的頻率。若燈號狀態正常，此時已進入 AT Mode
7. 為了驗證是否正確登入 AT Mode ，請開啟 Serial Moniter ，並設定視窗右下角的 Baud rate 為 9600，於 Serial Moniter 上方欄位中輸入大寫 "AT" ，將收到回應 "OK"



8. 如果依舊沒有辦法收到 " OK " 回覆，請再次檢查您的線路與步驟是否正確

HC-05 模組的 AT 指令需要加上\n\r

Serial Monitor: 

☞ AT+Version //查看韌體版本
+VERSION:2.0-20100601
OK

☞ AT+NAME? //傳回 BT 模組名稱
+NAME:HC-05
OK

☞ AT+PSWD? //傳回 BT 模組配對碼
+PSWD:1234
OK

☞ AT+UART? //查詢 BT 序列通訊參數，default: 9600,0(stop bit),0(parity bit)
+UART:9600,0,0
OK

☞ AT+UART=38400,1,0
OK

☞ AT+NAME=HIM00 //更改名稱為 HIM00
OK

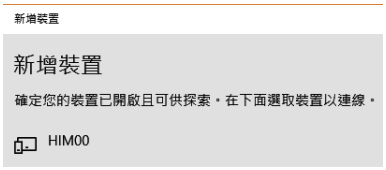
☞ AT+NAME? //傳回 BT 模組名稱
+NAME:HIM00
OK

在本實驗，每一組務必以組別編號來設定

驗證:(先拔掉 USB，再去除連接至 pin9 與 KEY/EN 的接線

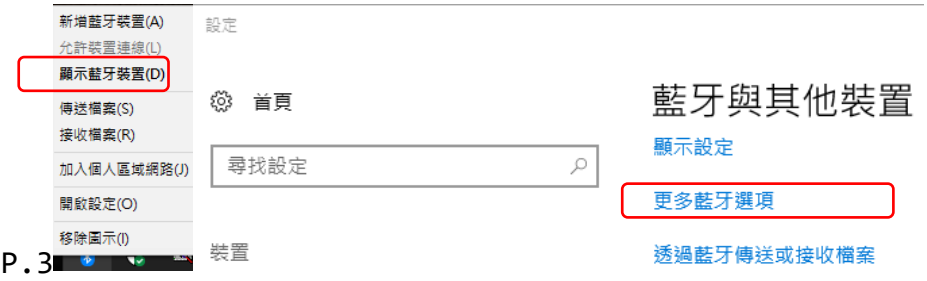
STEP.1

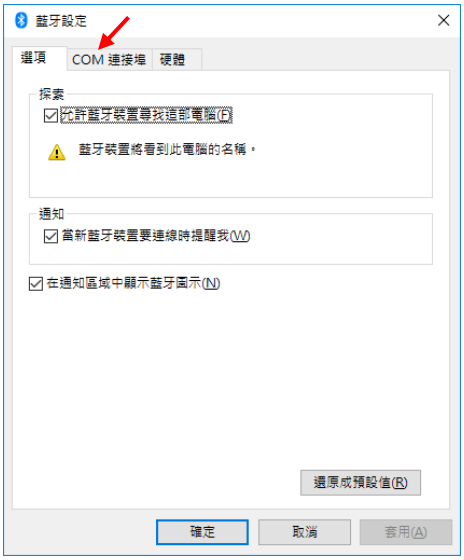
STEP.2



PW:1234

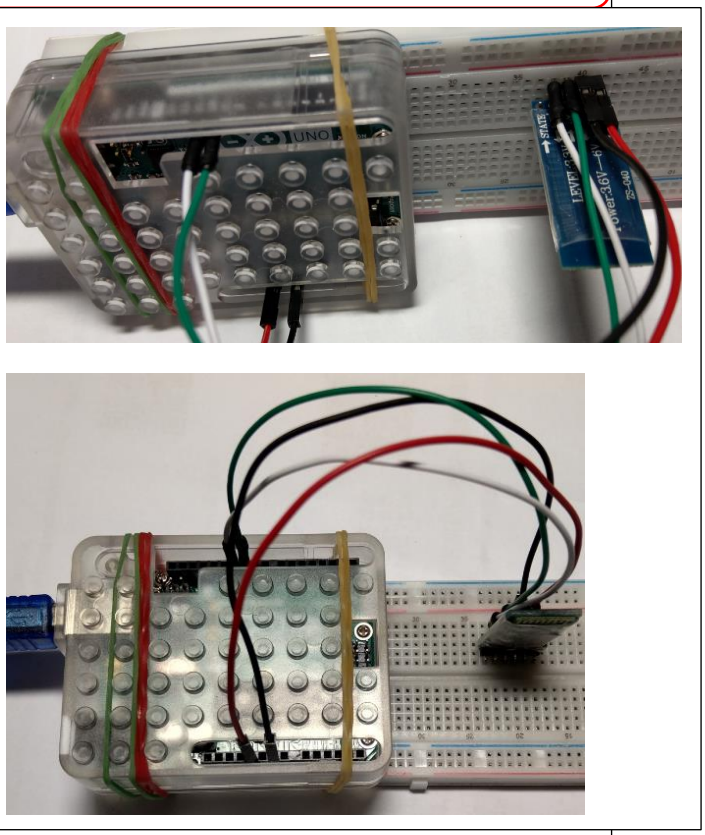
STEP.3

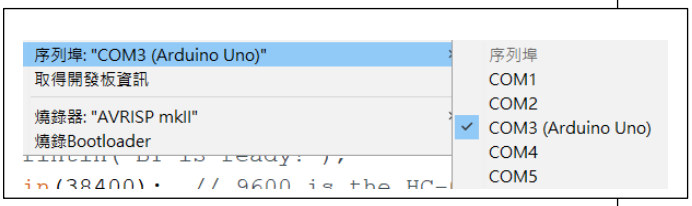


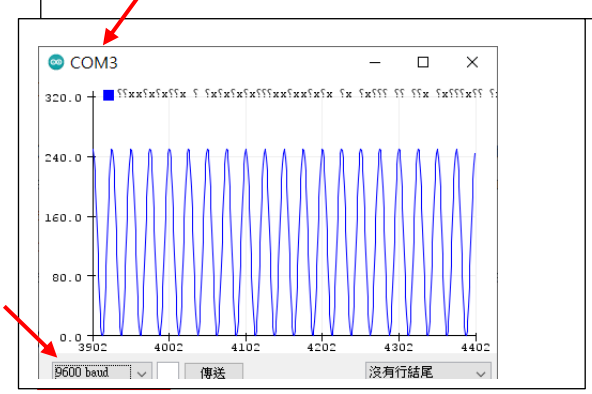


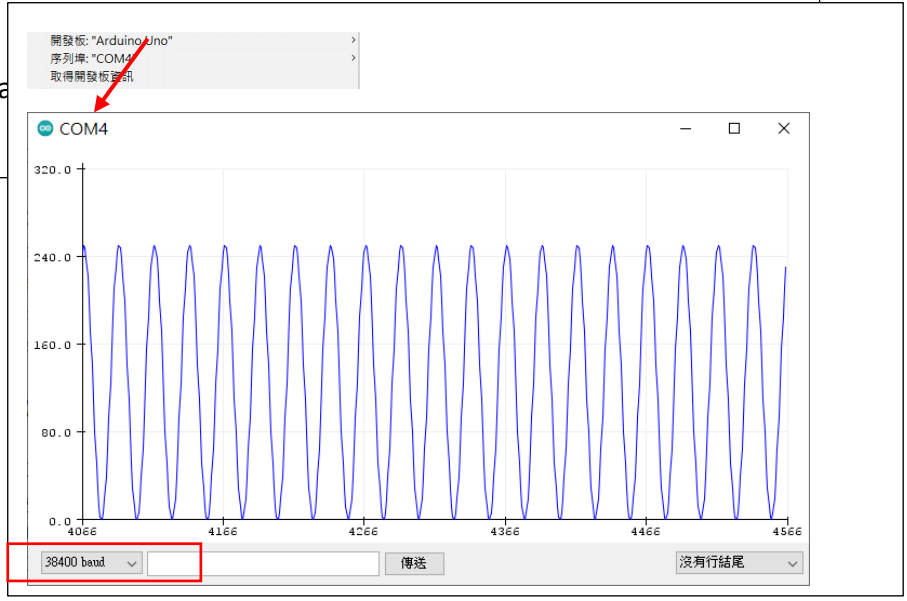
Ex5.02A Generating Sin Wave

```
#include <TimerOne.h>
#include <SoftwareSerial.h>
SoftwareSerial BT(10, 11); //Arduino's pin-10(Receiving):BT's Tx, Arduino's pin-11(Transmitting):BT's Rx
const byte LEDpin=13;
const int sampling=250, freq=10;
byte data;
float y;
void setup() {
   Serial.begin(9600);
   for (int i=0;i<5;i++)
   {
      digitalWrite(LEDpin,HIGH);
      delay(1000);
      digitalWrite(LEDpin,LOW);
   }
   Serial.println("Serial is ready!");
   BT.println("BT is ready!");
   BT.begin(38400);    // HC-05 default speed in AT comma
   // initialize timer1, and set a 4ms second period
   Timer1.initialize(4000);
   Timer1.attachInterrupt(callback);
   //attaches callback() as a timer overflow interrupt
}
void callback()
{
   y=125*sin(2*3.14159*freq*i++/sampling)+125;
   data=(byte)(y+0.5);
   //Serial.write(data)
   Serial.println(data);
   BT.println(data);
}
void loop() {
   // put your main code here, to run repea
}
```

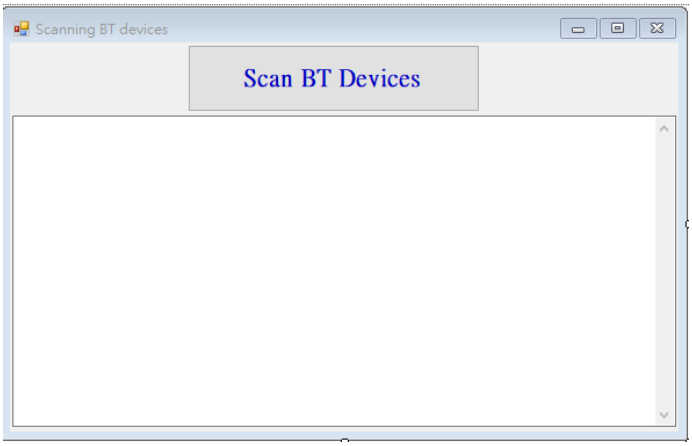


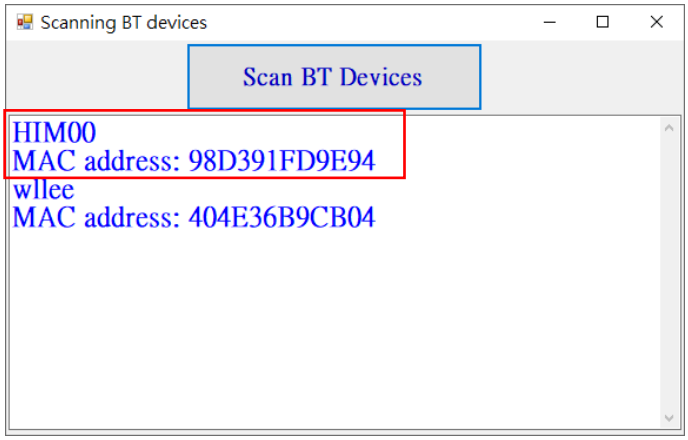






Ex5.03 Scanning BT Devices

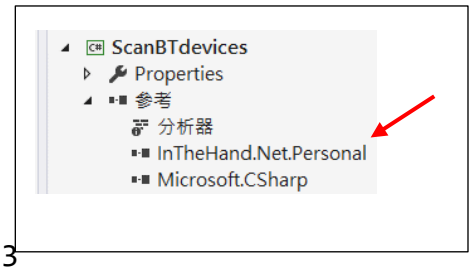


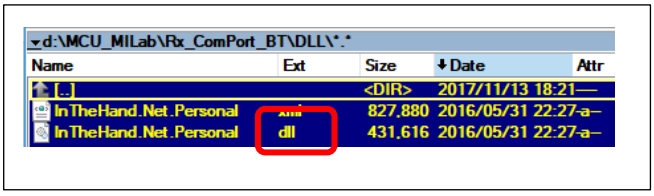


Step.1 Preparing BT dlls (download from https://inthehand.com/components/32feet/)

Creating a subdirectory to save theses dlls!

Step.2 Add References and namespace

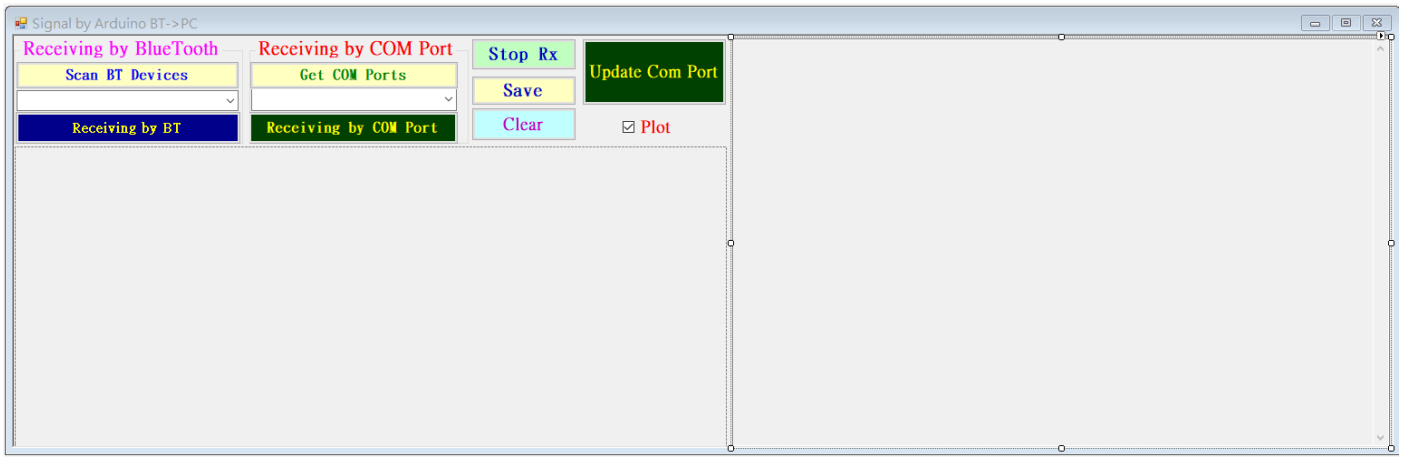




Step.3

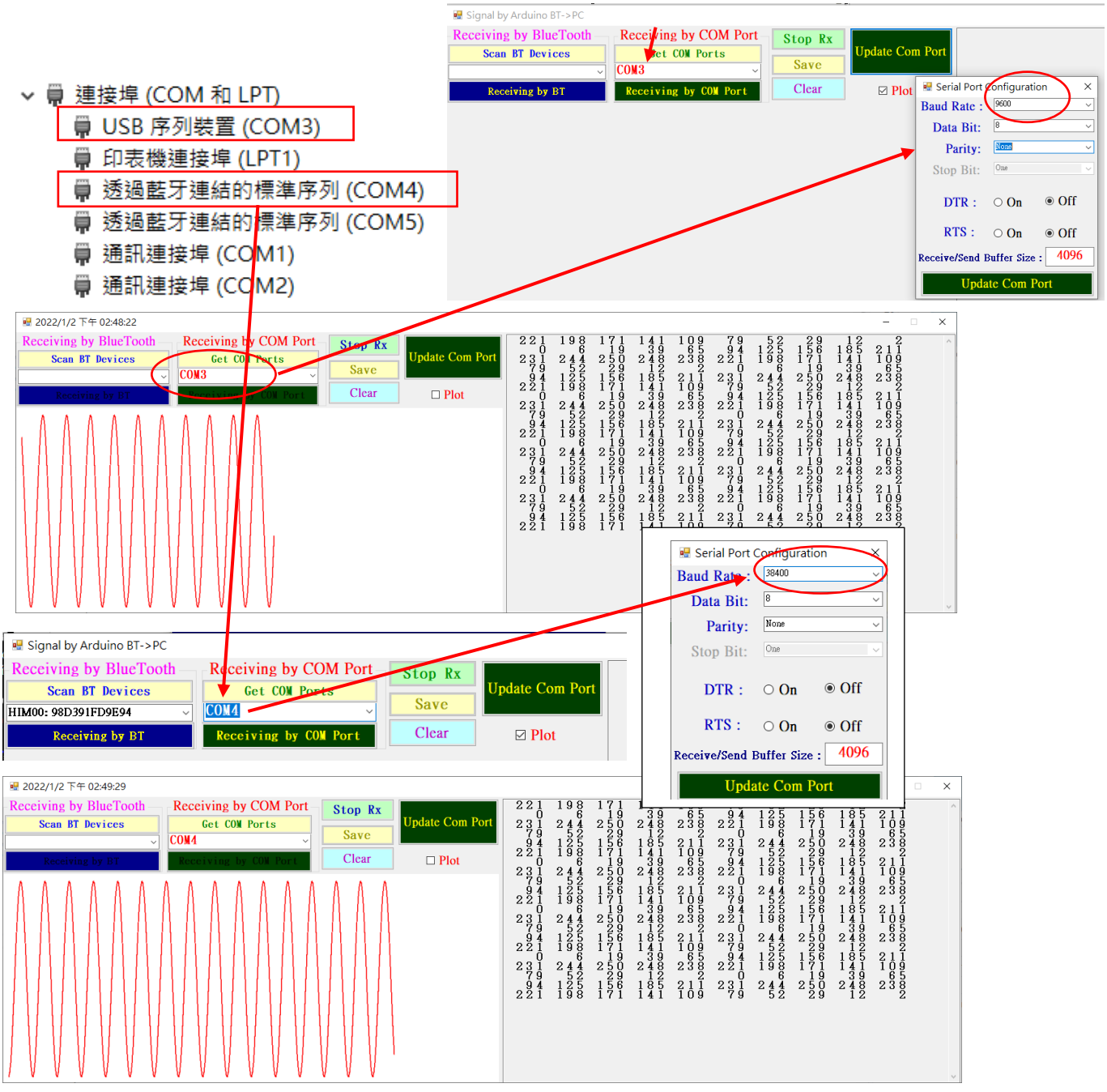
```
using System.Windows.Forms;
using InTheHand.Net;
using InTheHand.Net.Sockets;
using InTheHand.Net.Bluetooth;
namespace ScanBTdevices
{
    3 references
    public partial class Form1 : Form
    {
        1 reference
        public Form1()
        {
            InitializeComponent();
        }
        BluetoothAddress addr;   //用於識別一個獨特的藍牙裝置的地址
        BluetoothEndPoint ep;    //代表一個設備上的藍牙服務(端點)
        BluetoothClient cli;   //藍牙用戶端
        BluetoothAddress[] MACs;
        1 reference
        private void btnScan_Click(object sender, EventArgs e)
        {
            Cursor = Cursors.WaitCursor;
            BluetoothRadio.PrimaryRadio.Mode = RadioMode.Connectable;
            BluetoothClient client = new BluetoothClient();
            BluetoothDeviceInfo[] devices = client.DiscoverDevices();
            MACs = new BluetoothAddress[devices.Length];
            txtBxResult.Text = "";
            foreach (BluetoothDeviceInfo device in devices)
            {
                txtBxResult.Text += device.DeviceName + "\r\n";
                txtBxResult.Text += string.Format ("MAC address: {0}", device.DeviceAddress) + "\r\n";
            }
            Cursor = Cursors.Arrow;
        }
    }
}
```

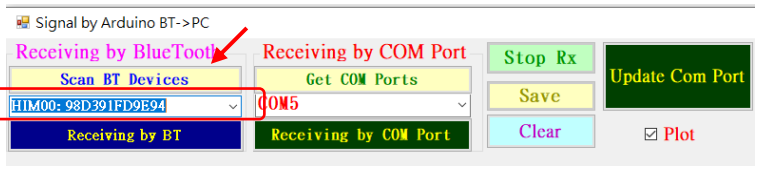
Ex5.04 Arduino_BT/COM2PC(Transfer the Sine Wave in Arduino by BT/COM to PC)



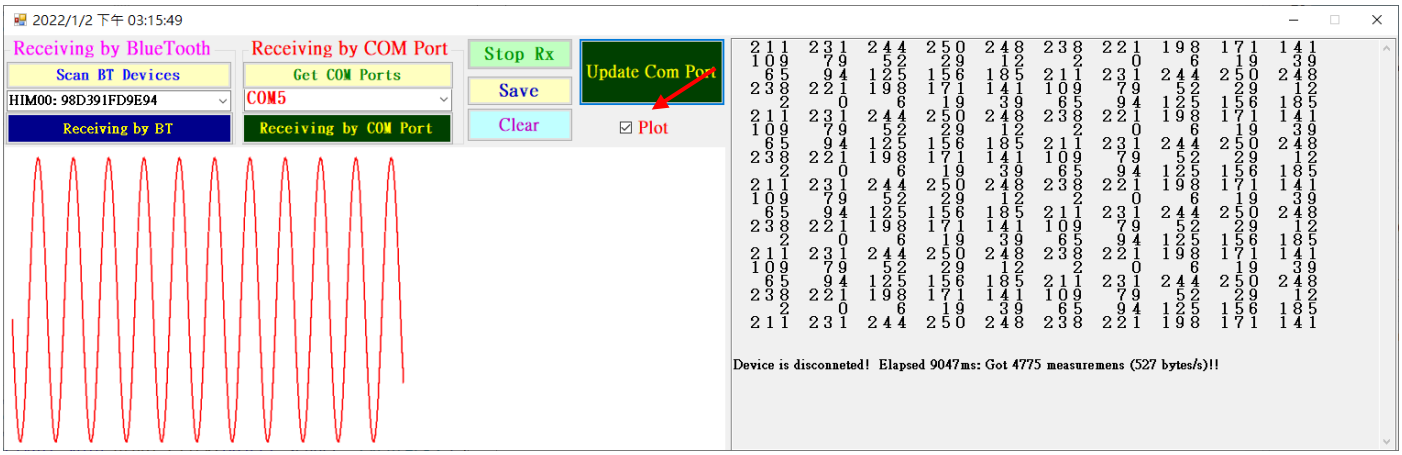
Test 3-Way: Arduino's COM Port, **BT's SPP**, and **BT(by third-party's dll)**

1. Arduino's COM Port

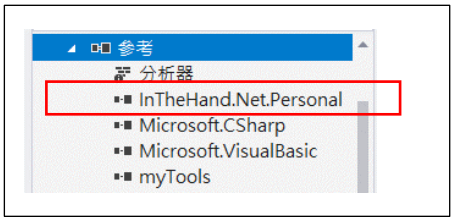




We don't need to select COM port and assign baud rate when using communication of Bluetooth.




```
using System.Windows.Forms;
using System.IO.Ports;
using System.Threading;
using System.IO;
using System.Diagnostics;
using mySerialPort;
using InTheHand.Net;
using InTheHand.Net.Sockets;
using InTheHand.Net.Bluetooth;
using myTools;
using Microsoft.VisualBasic;
```




```
namespace ArduinoBT2PC
{
    3 references
    public partial class MainForm1 : Form
    {
        //BT
        BluetoothAddress addr;//用於識別一個獨特的藍牙裝置的地址= BluetoothAddress.Parse("98:D3:32:10:B8:6F");//HC-05
                              //= BluetoothAddress.Parse("201408280542");//Appstudino
        BluetoothEndPoint ep; //代表一個設備上的藍牙服務(端點)
        BluetoothClient clnt; //藍牙用戶端
        BluetoothAddress[] MACs;
        Stream btStream;
        Thread RxBT;
        //
        delegate void DisplayG();
        DisplayG dispG;
        PortConfigForm setupComPort;
        StringBuilder res;
        int i,iStart,iEnd, iS, iE, LenRead, len,ii;
        List<byte> raw;
        byte[] RxBuf, buffer;
        const int bufLen = 2048;
        bool bBT, bCOM;
        Stopwatch sw;
        byte val;
        myWaveBMP myWave;
        Image img;
        1 reference
        public MainForm1()...
        1 reference
        private void displayG()...
        2 references
        private void initialStart()...
        2 references
        private void configPort()...
        1 reference
        private void serialPort1_DataReceived(object sender, SerialDataReceivedEventArgs e)...
        1 reference
        private void Form1_Load(object sender, EventArgs e)...
```

```
123     1 reference
        private void btnGet_Click(object sender, EventArgs e)...
127     1 reference
        private void btnUpdateComR_Click(object sender, EventArgs e)...
132     1 reference
        private void btnCOM_Click(object sender, EventArgs e)...
149     1 reference
        private void btnStop_Click(object sender, EventArgs e)...
167     1 reference
        private void timer1_Tick(object sender, EventArgs e)...
192     1 reference
        private void btnBT_Click(object sender, EventArgs e)...
202     Guid serviceClass;
203     1 reference
        void ConnectBtMedicalDevice()...
230     int i0;
231     1 reference
        private void ReceivingPacket()...
250     2 references
        void DisconnectBt()...
261     1 reference
        private void btnScan_Click(object sender, EventArgs e)...
280     1 reference
        private void Form1_FormClosing(object sender, FormClosingEventArgs e)...
286     1 reference
        private void btnSave_Click(object sender, EventArgs e)...
299     1 reference
        private void btnClear_Click(object sender, EventArgs e)...
304     }
305 }
```

```
46      public MainForm1()
47      {
48          InitializeComponent();
49          sw = new Stopwatch();
50      }
        1 reference
51      private void displayG()
52      {
53          iE = raw.Count - 1;
54          while (iS <= iE)
55          {
56              val = raw[iS++];
57              myWave.update(val);
58          }
59          if (img != null)
60          {
61              img.Dispose();
62              img = null;
63          }
64          img = myWave.getBMP();
65          pictureBox1.Image = img;
66      }
```

```
67      private void initialStart()
68      {
69          btnStop.Enabled = true;
70          btnBT.Enabled = false;
71          btnSave.Enabled = false;
72          btnCOM.Enabled = false;
73          iS = 0;
74          iE = 0;
75          myWave = new myWaveBMP(500);//show 2-period since sampling rate:250
76          if (img != null)
77          {
78              img.Dispose();
79              img = null;
80          }
81          img = myWave.getBMP();
82          pictureBox1.Image = img;
83          raw.Clear();
84          sw.Restart();
85      }
```

```
86      private void configPort()
87      {
88          cmboBxPort.Items.Clear();
89          string[] ports = SerialPort.GetPortNames();
90          Array.Sort(ports);
91          foreach (string port in ports)
92          {
93              if (cmboBxPort.Items.Count > 0 && cmboBxPort.Items[cmboBxPort.Items.Count - 1].ToString().Contains(port))
94                  continue;
95              cmboBxPort.Items.Add(port);
96          }
97          cmboBxPort.SelectedIndex = cmboBxPort.Items.Count - 1;
98          btnCOM.Enabled = true;
99          serialPort1.PortName = cmboBxPort.SelectedItem.ToString();
100     }
        1 reference
101     private void serialPort1_DataReceived(object sender, SerialDataReceivedEventArgs e)
102     {
103         if (bCOM && serialPort1.BytesToRead > 0)
104         {
105             len = serialPort1.Read(buffer, 0, buffer.Length);
106             for (int i = 0; i < len; i++)
107                 raw.Add(buffer[i]);
108             if (checkBox1.Checked)
109                 BeginInvoke(dispG, new Object[] { });
110         }
111     }
```

```
112     private void Form1_Load(object sender, EventArgs e)
113     {
114         dispG = new DisplayG(displayG);
115         raw = new List<byte>();
116         configPort();
117         res = new StringBuilder();
118         setupComPort = new PortConfigForm(ref serialPort1);
119         //setupComPort.ShowDialog();
120         bBT = false;
121         bCOM = false;
122     }
        1 reference
123     private void btnGet_Click(object sender, EventArgs e)
124     {
125         configPort();
126     }
        1 reference
127     private void btnUpdateComR_Click(object sender, EventArgs e)
128     {
129         setupComPort.ComPortConfig(ref serialPort1);
130         setupComPort.ShowDialog();
131     }
```

```
        1 reference
132     private void btnCOM_Click(object sender, EventArgs e)
133     {
134         if (bBT)
135             DisconnectBt();
136         bBT = false;
137         bCOM = true;
138         serialPort1.PortName = cmboBxPort.SelectedItem.ToString();
139
140         if (buffer != null)//for COM
141             buffer = null;
142         buffer = new byte[serialPort1.ReadBufferSize];
143         initialStart();
144         ii = 0;
145         serialPort1.Open();
146         Text = "Device is connected";
147         timer1.Start();
148     }
```

```
149         private void btnStop_Click(object sender, EventArgs e)
150         {
151             bBT = false;
152             bCOM = false;
153             btnCOM.Enabled = true;
154             btnBT.Enabled = true;
155             sw.Stop();
156             txtBxR.Text += string.Format("\r\n\r\nDevice is disconneted!  Elapsed {0}ms: Got {1} measuremens ({2} bytes/s)!!",
157                     sw.ElapsedMilliseconds, raw.Count, raw.Count * 1000 / sw.ElapsedMilliseconds);
158             Thread.Sleep(1000);//To avoid crush, 01/02/2021
159             btnStop.Enabled = false;
160             btnSave.Enabled = true;
161             if (serialPort1.IsOpen)
162                 serialPort1.Close();
163             DisconnectBt();
164             Application.DoEvents();
165             timer1.Stop();
166         }
167         private void timer1_Tick(object sender, EventArgs e)
168         {//顯示正在接收資料
169             Text = DateTime.Now.ToString();
170             if (checkBox1.Checked)
171                 return;
172             iStart = iEnd;
173             iEnd = raw.Count;
174             while (iStart < iEnd)
175             {
176                 //res.AppendFormat("{0:d3} ", raw[iStart++]);
177                 res.Append(Strings.StrConv(raw[iStart++].ToString(), VbStrConv.Wide, 0x0404).PadLeft(4, '  '));
178                 if (iStart % 10 == 0)
179                 {
180                     res.AppendLine();
181                     ii++;
182                     if (ii > 20)
183                     {
184                         txtBxR.Text= res.ToString();
185                         res.Clear();
186                         ii = 0;
187                     }
188                 }
189             }
190             Application.DoEvents();
191         }
            1 reference
192         private void btnBT_Click(object sender, EventArgs e)
193         {//connect by BT
194             if (serialPort1.IsOpen)
195                 serialPort1.Close();
196             bBT = true;
197             bCOM = false;
198             initialStart();
199             ConnectBtMedicalDevice();
200             timer1.Start();
201         }
```

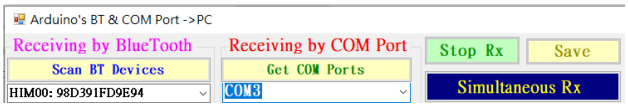
```
202         Guid serviceClass;
            1 reference
203         void ConnectBtMedicalDevice()
204         {
205             try
206             {
207                 serviceClass = BluetoothService.SerialPort;
208                 addr = MACs[cmbBxBT.SelectedIndex];
209                 if (ep != null)
210                     ep = null;
211                 ep = new BluetoothEndPoint(addr, serviceClass);//BluetoothService.SerialPort);
212                 if (clnt != null)
213                 {
214                     clnt.Close();
215                     clnt.Dispose();
216                     clnt = null;
217                 }
218                 clnt= new BluetoothClient();
219                 clnt.Connect(ep);
220                 btStream = clnt.GetStream();
221                 RxBuf = new byte[bufLen];
222                 RxBT = new Thread(ReceivingPacket);
223                 RxBT.Start();
224             }
225             catch (Exception ex)
226             {
227                 MessageBox.Show(ex.Message, "Error: Connecting BlueTooth");
228             }
229         }

230         int i0;
            1 reference
231         private void ReceivingPacket()
232         {//using thread to receive stream from BT
233             while (bBT)
234             {
235                 LenRead = 0;
236                 if (!btStream.CanRead)
237                     break;
238                 else if (bBT)
239                     LenRead = btStream.Read(RxBuf, 0, bufLen);   //資料讀取
240                 else
241                     break;
242                 if (LenRead == 0)
243                     continue;
244                 for (i0 = 0; i0 < LenRead; i0++)
245                     raw.Add(RxBuf[i0]);
246                 if (checkBox1.Checked)
247                     BeginInvoke(dispG, new Object[] { });
248             }
249         }
            2 references
250         void DisconnectBt()
251         {
252             if (clnt != null)
253             {
254                 clnt.Close();
255                 clnt.Dispose();
256                 clnt = null;
257             }
258             if (RxBT != null)
259                 RxBT.Join();
260         }
```

```
261         private void btnScan_Click(object sender, EventArgs e)
262         {
263             Cursor = Cursors.WaitCursor;
264             BluetoothRadio.PrimaryRadio.Mode = RadioMode.Connectable;
265             BluetoothClient client = new BluetoothClient();
266             BluetoothDeviceInfo[] devices = client.DiscoverDevices();
267             MACs = new BluetoothAddress[devices.Length];
268             i = 0;
269             cmbBxBT.Items.Clear();
270             foreach (BluetoothDeviceInfo device in devices)
271             {
272                 cmbBxBT.Items.Add(string.Format("{0}: {1}", device.DeviceName, device.DeviceAddress));
273                 MACs[i++] = device.DeviceAddress;
274             }
275             cmbBxBT.SelectedIndex = 0;
276             Cursor = Cursors.Arrow;
277             if (devices.Length > 0)
278                 btnBT.Enabled = true;
279         }
                1 reference
280         private void Form1_FormClosing(object sender, FormClosingEventArgs e)
281         {
282             if(serialPort1.IsOpen)
283                 serialPort1.Close();
284             serialPort1.Dispose();
285         }

                1 reference
286         private void btnSave_Click(object sender, EventArgs e)
287         {
288             serialPort1.Close();
289             saveFileDialog1.FileName = string.Format("BT2PC_{0:D4}{1:D2}{2:D2}_{3:D2}{4:D2}{5:D2}_{6}ms.txt",
290                         DateTime.Now.Year, DateTime.Now.Month, DateTime.Now.Day,
291                         DateTime.Now.Hour, DateTime.Now.Minute, DateTime.Now.Second,sw.ElapsedMilliseconds);
292             if (saveFileDialog1.ShowDialog() != System.Windows.Forms.DialogResult.OK)
293                 return;
294             StringBuilder sb = new StringBuilder();
295             for (int i = 0; i < raw.Count; i++)
296                 sb.AppendLine(raw[i].ToString());
297             File.AppendAllText(saveFileDialog1.FileName, sb.ToString());
298         }
                1 reference
299         private void btnClear_Click(object sender, EventArgs e)
300         {
301             txtBxR.Text = "";
302             raw.Clear();
303         }
304     }
305 }
```

Ex5_05 Arduino transmit data to PC by BT and COM port simultaneously




```
using System.IO.Ports;
using System.Threading;
using System.IO;
using System.Diagnostics;
using mySerialPort;
using InTheHand.Net;
using InTheHand.Net.Sockets;
using InTheHand.Net.Bluetooth;
using myTools;
using Microsoft.VisualBasic;
namespace ArduinoBT2PC_Sync
{
    - 參考
    public partial class MainForm1 : Form
    {
        //BT
        BluetoothAddress addr;  //用於識別一個獨特的藍牙裝置的地址= BluetoothAddress.Parse("98:D3:32:10:B8:6F");//HC-05
                                //= BluetoothAddress.Parse("201408280542");//Appstudino
        BluetoothEndPoint ep;   //代表一個設備上的藍牙服務(端點)
        BluetoothClient clnt;    //藍牙用戶端
        BluetoothAddress[] MACs;
        Stream btStream;
        Thread RxBT;
        //
        delegate void DisplayG();
        DisplayG dispG, dispG2;
        PortConfigForm setupComPort;
        int i,iS2,iE2, iS, iE, LenRead, len;
        List<byte> raw,raw2;
        byte[] RxBuf, buffer;
        const int bufLen = 2048;
        bool bRx;
        Stopwatch sw;
```

```
42          byte val;
43          myWaveBMP myWave, myWave2;
44          Image img, img2;
            - 参考
45          public MainForm1()...
            - 参考
50          private void displayG()...
            - 参考
66          private void displayG2()...
            - 参考
82          private void initialStart()...
            - 参考
111         private void configPort()...
            - 参考
125         private void serialPort1_DataReceived(object sender, SerialDataReceivedEventArgs e)...
            - 参考
135         private void Form1_Load(object sender, EventArgs e)...
            - 参考
146         private void btnGet_Click(object sender, EventArgs e)...
            - 参考
150         private void btnUpdateComR_Click(object sender, EventArgs e)...
            - 参考
155         private void btnStop_Click(object sender, EventArgs e)...
            - 参考
170         private void timer1_Tick(object sender, EventArgs e)...

176         private void btnRx_Click(object sender, EventArgs e)...
190         Guid serviceClass;
            - 参考
191         void ConnectBtMedicalDevice()...
218         int i0;
            - 参考
219         private void ReceivingPacket()...
            - 参考
245         void DisconnectBt()...
            - 参考
256         private void btnScan_Click(object sender, EventArgs e)...
            - 参考
275         private void Form1_FormClosing(object sender, FormClosingEventArgs e)...
            - 参考
281         private void btnSave_Click(object sender, EventArgs e)...
294     }
295 }
```

```
45          public MainForm1()
46          {
47              InitializeComponent();
48              sw = new Stopwatch();
49          }
            - 参考
50          private void displayG()
51          {
52              iE = raw.Count - 1;
53              while (bRx && iS <= iE)
54              {
55                  val = raw[iS++];
56                  myWave.update(val);
57              }
58              if (img != null)
59              {
60                  img.Dispose();
61                  img = null;
62              }
63              img = myWave.getBMP();
64              pictureBox1.Image = img;
65          }
```

```
66          private void displayG2()
67          {
68              iE2 = raw2.Count - 1;
69              while (bRx && iS2 <= iE2)
70              {
71                  val = raw2[iS2++];
72                  myWave2.update(val);
73              }
74              if (img2 != null)
75              {
76                  img2.Dispose();
77                  img2 = null;
78              }
79              img2 = myWave2.getBMP();
80              pictureBox2.Image = img2;
81          }
```

```
82          private void initialStart()
83          {
84              btnStop.Enabled = true;
85              btnRx.Enabled = false;
86              btnSave.Enabled = false;
87              iS = 0;
88              iE = 0;
89              iS2 = 0;
90              iE2 = 0;
91              myWave = new myWaveBMP(500);//show 2-period since sampling rate:250
92              myWave2 = new myWaveBMP(500);
93              if (img != null)
94              {
95                  img.Dispose();
96                  img = null;
97              }
98              img = myWave.getBMP();
99              pictureBox1.Image = img;
100             if (img2 != null)
101             {
102                 img2.Dispose();
103                 img2 = null;
104             }
105             img2 = myWave2.getBMP();
106             pictureBox2.Image = img2;
107             raw.Clear();
108             raw2.Clear();
109             sw.Restart();
110         }
```

```
111     private void configPort()
112     {
113         cmboBxPort.Items.Clear();
114         string[] ports = SerialPort.GetPortNames();
115         Array.Sort(ports);
116         foreach (string port in ports)
117         {
118             if (cmboBxPort.Items.Count > 0 && cmboBxPort.Items[cmboBxPort.Items.Count - 1].ToString().Contains(port))
119                 continue;
120             cmboBxPort.Items.Add(port);
121         }
122         cmboBxPort.SelectedIndex = cmboBxPort.Items.Count - 1;
123         serialPort1.PortName = cmboBxPort.SelectedItem.ToString();
124     }
        參考
125     private void serialPort1_DataReceived(object sender, SerialDataReceivedEventArgs e)
126     {
127         if (bRx && serialPort1.BytesToRead > 0)
128         {
129             len = serialPort1.Read(buffer, 0, buffer.Length);
130             for (int i = 0; i < len; i++)
131                 raw.Add(buffer[i]);
132             BeginInvoke(dispG, new Object[] { });
133         }
134     }
```

```
135         private void Form1_Load(object sender, EventArgs e)
136         {
137             dispG = new DisplayG(displayG);
138             dispG2 = new DisplayG(displayG2);
139             raw = new List<byte>();
140             raw2 = new List<byte>();
141             configPort();
142             setupComPort = new PortConfigForm(ref serialPort1);
143             //setupComPort.ShowDialog();
144             bRx = false;
145         }
146         private void btnGet_Click(object sender, EventArgs e)
147         {
148             configPort();
149         }
150         private void btnUpdateComR_Click(object sender, EventArgs e)
151         {
152             setupComPort.ComPortConfig(ref serialPort1);
153             setupComPort.ShowDialog();
154         }

155         private void btnStop_Click(object sender, EventArgs e)
156         {
157             bRx = false;
158             btnRx.Enabled = true;
159             btnStop.Enabled = false;
160             btnSave.Enabled = true;
161             if (serialPort1.IsOpen)
162                 serialPort1.Close();
163             DisconnectBt();
164             sw.Stop();
165             timer1.Stop();
166             Text += string.Format("Device is disconneted!  Elapsed {0}ms: Got {1} measuremens ({2} bytes/s)!!",
167                     sw.ElapsedMilliseconds, raw.Count, raw.Count * 1000 / sw.ElapsedMilliseconds);
168             Application.DoEvents();
169         }
170         private void timer1_Tick(object sender, EventArgs e)
171         {//顯示正在接收資料
172             Text = DateTime.Now.ToString();
173             //displayG2();
174             Application.DoEvents();
175         }

176         private void btnRx_Click(object sender, EventArgs e)
177         {
178             serialPort1.PortName = cmboBxPort.SelectedItem.ToString();
179             if (buffer != null)//for COM
180                 buffer = null;
181             buffer = new byte[serialPort1.ReadBufferSize];
182             initialStart();
183             //connect by BT
184             bRx = true;
185             ConnectBtMedicalDevice();
186             serialPort1.Open();
187             Text = "Device is connected";
188             timer1.Start();
189         }
```

```
190     Guid serviceClass;
        - 參考
191     void ConnectBtMedicalDevice()
192     {
193         try
194         {
195             serviceClass = BluetoothService.SerialPort;
196             addr = MACs[cmbBxBT.SelectedIndex];
197             if (ep != null)
198                 ep = null;
199             ep = new BluetoothEndPoint(addr, serviceClass);//BluetoothService.SerialPort);
200             if (clnt != null)
201             {
202                 clnt.Close();
203                 clnt.Dispose();
204                 clnt = null;
205             }
206             clnt= new BluetoothClient();
207             clnt.Connect(ep);
208             btStream = clnt.GetStream();
209             RxBuf = new byte[bufLen];
210             RxBT = new Thread(ReceivingPacket);
211             RxBT.Start();
212         }
213         catch (Exception ex)
214         {
215             MessageBox.Show(ex.Message, "Error: Connecting BlueTooth");
216         }
217     }
```

```
218     int i0;
        - 參考
219     private void ReceivingPacket()
220     {//using thread to receive stream from BT
221         try
222         {
223             while (bRx)
224             {
225                 LenRead = 0;
226                 if (btStream.CanRead)
227                     LenRead = btStream.Read(RxBuf, 0, bufLen);    //資料讀取
228                 if (LenRead == 0)
229                     continue;
230                 for (i0 = 0; i0 < LenRead; i0++)
231                     raw2.Add(RxBuf[i0]);
232                 BeginInvoke(dispG2, new Object[] { });
233             }
234         }
235         catch(Exception ex)
236         {
237             Text = ex.ToString();
238         }
239         finally
240         {
241
242         }
243
244     }
```

```
245         void DisconnectBt()
246         {
247             if (clnt != null)
248             {
249                 clnt.Close();
250                 clnt.Dispose();
251                 clnt = null;
252             }
253             if (RxBT != null)
254                 RxBT.Join();
255         }
            参考
256         private void btnScan_Click(object sender, EventArgs e)
257         {
258             Cursor = Cursors.WaitCursor;
259             BluetoothRadio.PrimaryRadio.Mode = RadioMode.Connectable;
260             BluetoothClient client = new BluetoothClient();
261             BluetoothDeviceInfo[] devices = client.DiscoverDevices();
262             MACs = new BluetoothAddress[devices.Length];
263             i = 0;
264             cmbBxBT.Items.Clear();
265             foreach (BluetoothDeviceInfo device in devices)
266             {
267                 cmbBxBT.Items.Add(string.Format("{0}: {1}", device.DeviceName, device.DeviceAddress));
268                 MACs[i++] = device.DeviceAddress;
269             }
270             cmbBxBT.SelectedIndex = 0;
271             Cursor = Cursors.Arrow;
272             if (devices.Length > 0)
273                 btnRx.Enabled = true;
274         }
            参考
275         private void Form1_FormClosing(object sender, FormClosingEventArgs e)
276         {
277             if(serialPort1.IsOpen)
278                 serialPort1.Close();
279             serialPort1.Dispose();
280         }
            参考
281         private void btnSave_Click(object sender, EventArgs e)
282         {
283             serialPort1.Close();
284             saveFileDialog1.FileName = string.Format("BT_COM2PC_{0:D4}{1:D2}{2:D2}_{3:D2}{4:D2}{5:D2}_{6}ms.txt",
285                         DateTime.Now.Year, DateTime.Now.Month, DateTime.Now.Day,
286                         DateTime.Now.Hour, DateTime.Now.Minute, DateTime.Now.Second,sw.ElapsedMilliseconds);
287             if (saveFileDialog1.ShowDialog() != System.Windows.Forms.DialogResult.OK)
288                 return;
289             StringBuilder sb = new StringBuilder();
290             for (int i = 0; i < raw.Count; i++)
291                 sb.AppendLine(raw[i].ToString());
292             File.AppendAllText(saveFileDialog1.FileName, sb.ToString());
293         }
294     }
295 }
```