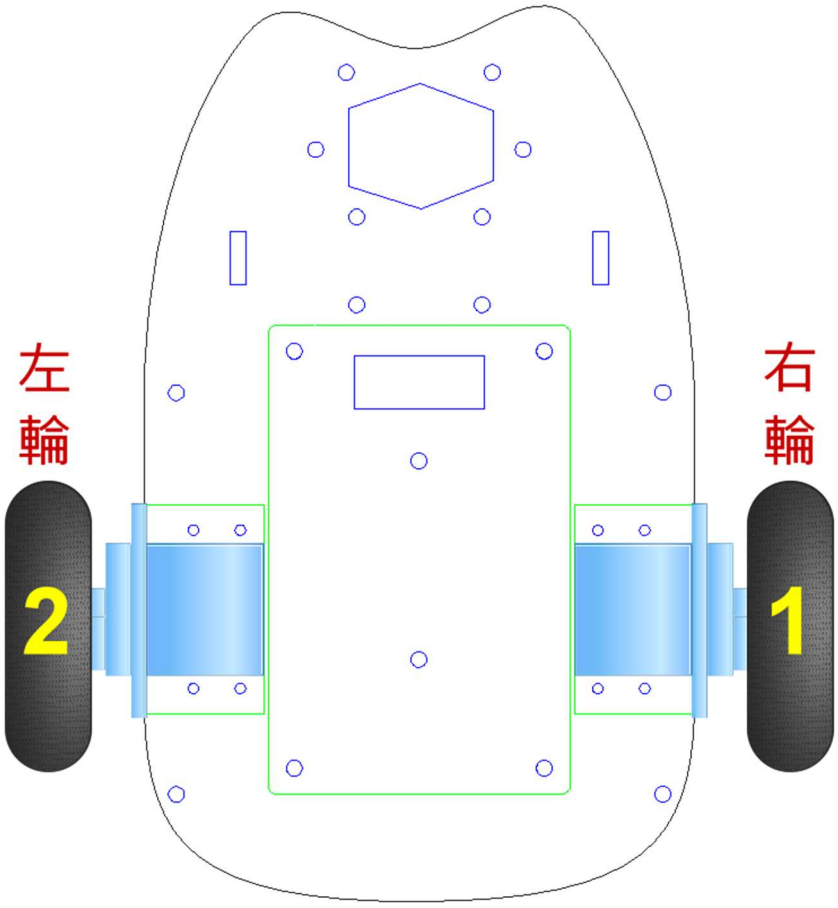
車體控制



360 度伺服馬達控制方式

右轉 → 1 順時針 2 順時針

左轉 → 1 逆時針 2 逆時針

前進 → 1 逆時針 2 順時針

後退 → 1 順時針 2 逆時針


設定
伺服馬達 腳位# 3
角度(0~180) 130
延遲 0
360度伺服馬達 腳位# 10 停止轉動
360度伺服馬達 腳位# 11 停止轉動
設定串列埠 serial 傳輸率 9600 bps
宣告 Data 當 int 資料 0
迴圈
如果 串列埠有效資料? > 0
執行 賦值 Data 到 串列埠輸入
如果 Data = 'F'
執行
360度伺服馬達 腳位# 10
旋轉方向 逆時針
速度 快速
360度伺服馬達 腳位# 11
旋轉方向 順時針
速度 快速
否則如果 Data = 'B'


否則如果 Data = 'B'
執行
360度伺服馬達 腳位# 10
旋轉方向 順時針
速度 快速
360度伺服馬達 腳位# 11
旋轉方向 逆時針
速度 快速
否則如果 Data = 'L'
執行
360度伺服馬達 腳位# 10
旋轉方向 順時針
速度 快速
360度伺服馬達 腳位# 11
旋轉方向 順時針
速度 快速
否則如果 Data = 'R'


否則如果 Data = 'R'
執行
360度伺服馬達 腳位# 10
旋轉方向 逆時針
速度 快速
360度伺服馬達 腳位# 11
旋轉方向 逆時針
速度 快速
否則如果 Data = 'S'
執行
360度伺服馬達 腳位# 10 停止轉動
360度伺服馬達 腳位# 11 停止轉動
否則如果 Data = 'U'


否則如果 Data = 'U'
執行
伺服馬達 腳位# 3
角度(0~180) 180
延遲 0
否則如果 Data = 'D'
執行
伺服馬達 腳位# 3
角度(0~180) 130
延遲 0

原始程式

```
/* Sketch was generated by motoblockly
 Website: http://www.motoblockly.com
 Author: www.motoduino.com
 Date: Fri Sep 17 2021 12:23:34 GMT+0800
 */
#include <Servo.h>

Servo servo_3;

int  Data;
Servo servo_10;

Servo servo_11;

void setup()
{
  Serial.begin(9600);
  servo_10.attach(10);

  servo_11.attach(11);

    servo_3.attach(3);
  servo_3.write(130);
  delay(0);
  servo_10.write(90);
  servo_11.write(90);
  Data = 0;

}


void loop()
{
    if (Serial.available() > 0) {
      Data = Serial.read();
    }
    if (Data == 'F') {
      servo_10.write(140);
      servo_11.write(40);
    } else if (Data == 'B') {
```

```
      servo_10.write(40);
      servo_11.write(140);
    } else if (Data == 'L') {
      servo_10.write(40);
      servo_11.write(40);
    } else if (Data == 'R') {
      servo_10.write(140);
      servo_11.write(140);
    } else if (Data == 'S') {
      servo_10.write(90);
      servo_11.write(90);
    } else if (Data == 'U') {
      servo_3.attach(3);
      servo_3.write(180);
      delay(0);
    } else if (Data == 'D') {
      servo_3.attach(3);
      servo_3.write(130);
      delay(0);
    }

}
```

程式修正

```
#include <SoftwareSerial.h>       //新增
#include <Servo.h>
SoftwareSerial BT(6,7);           //新增
Servo servo_3;

int    Data;
Servo servo_10;

Servo servo_11;

void setup()
{
   BT.begin(9600);                //更名
   servo_10.attach(10);
   servo_11.attach(11);
```

```
    servo_3.attach(3);
    servo_3.write(90);
    delay(0);
    servo_10.write(90);
    servo_11.write(90);
    Data = 0;

}


void loop()
{
      if (BT.available() > 0) {              //更名
         Data = BT.read();                   //更名
      }
      if (Data == 'F') {
         servo_10.write(140);
         servo_11.write(40);
      } else if (Data == 'B') {
         servo_10.write(40);
         servo_11.write(140);
      } else if (Data == 'L') {
         servo_10.write(40);
         servo_11.write(40);
      } else if (Data == 'R') {
         servo_10.write(140);
         servo_11.write(140);
      } else if (Data == 'S') {
         servo_10.write(90);
         servo_11.write(90);
      } else if (Data == 'U') {
         servo_3.attach(3);
         servo_3.write(180);
         delay(0);
      } else if (Data == 'D') {
         servo_3.attach(3);
         servo_3.write(130);
         delay(0);
      }

}
```

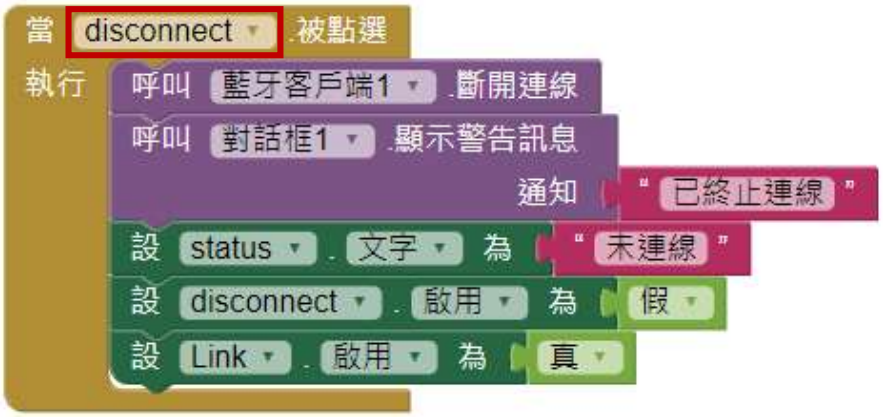
APPInverter 程式

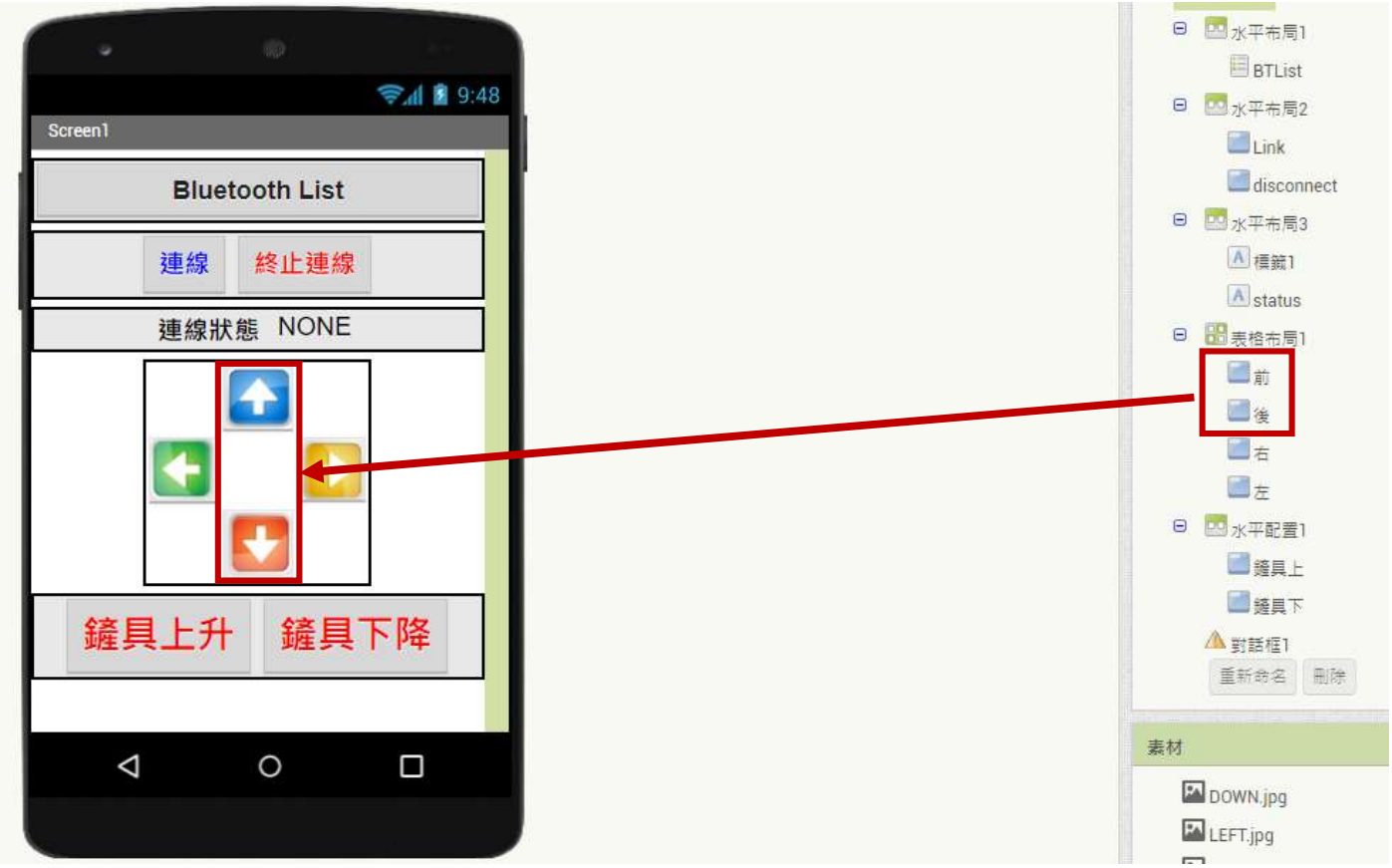

組件列表
Screen1
水平布局1
BTList
水平布局2
Link
disconnect
水平布局3
標籤1
status
表格布局1
前
後
右
左
水平配置1
鏟具上
鏟具下
對話框1
重新命名
刪除
素材
DOWN.jpg
LEFT.jpg
RIGHT.jpg
UP.jpg
上傳文件...
顯示隱藏組件
9:48
Screen1
Bluetooth List
連線
終止連線
連線狀態 NONE
鏟具上升
鏟具下降
非可視組件
對話框1
藍牙客戶端1

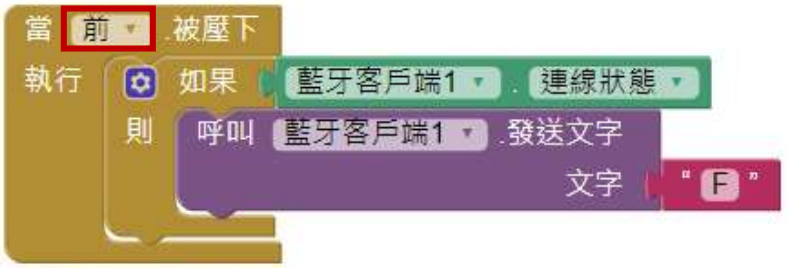

當 BTList .準備選擇
執行 設 BTList . 元素 為 藍牙客戶端1 . 位址及名稱
呼叫 藍牙客戶端1 .斷開連線
當 BTList .選擇完成
執行 設置 全域 BTdevice 為 BTList . 選中項
設 BTList . 文字 為 取得 全域 BTdevice
呼叫 對話框1 .顯示警告訊息
通知 " 請按連線按鈕進行藍牙連接 "
設 status . 文字 為 " 請按連線按鈕 "

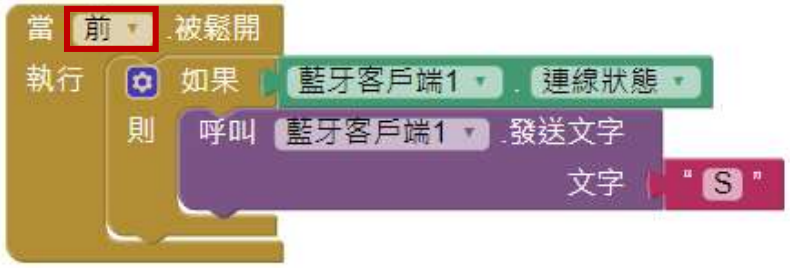

9:48
Screen1
Bluetooth List
連線
終止連線
連線狀態 NONE
鏟具上升
鏟具下降
水平布局1
BTList
水平布局2
Link
disconnect
水平布局3
標籤1
status
表格布局1
前
後
右
左
水平配置1
鏟具上
鏟具下
對話框1
重新命名
刪除
素材
DOWN.jpg
LEFT.jpg

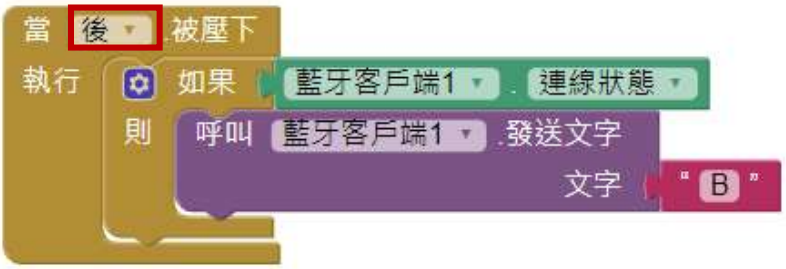

當 Link .被點選
執行 如果 呼叫 藍牙客戶端1 .連線
地址 取得 全域 BTdevice
則 設 Link . 啟用 為 假
設 disconnect . 啟用 為 真
呼叫 對話框1 .顯示警告訊息
通知 " 已連線 "
設 status . 文字 為 " 已連線 "
否則 呼叫 對話框1 .顯示警告訊息
通知 " 未連線 "
設 status . 文字 為 " 未連線 "


當 disconnect .被點選
執行 呼叫 藍牙客戶端1 .斷開連線
呼叫 對話框1 .顯示警告訊息
通知 " 已終止連線 "
設 status . 文字 為 " 未連線 "
設 disconnect . 啟用 為 假
設 Link . 啟用 為 真


9:48
Screen1
Bluetooth List
連線
終止連線
連線狀態 NONE
鏟具上升
鏟具下降
水平布局1
BTList
水平布局2
Link
disconnect
水平布局3
標籤1
status
表格布局1
前
後
右
左
水平配置1
鏟具上
鏟具下
對話框1
重新命名
刪除
素材
DOWN.jpg
LEFT.jpg


當 前 .被壓下
執行
如果 藍牙客戶端1 . 連線狀態
則
呼叫 藍牙客戶端1 .發送文字
文字 " F "


當 前 .被鬆開
執行
如果 藍牙客戶端1 . 連線狀態
則
呼叫 藍牙客戶端1 .發送文字
文字 " S "


當 後 .被壓下
執行
如果 藍牙客戶端1 . 連線狀態
則
呼叫 藍牙客戶端1 .發送文字
文字 " B "


當 後 .被鬆開
執行
如果 藍牙客戶端1 . 連線狀態
則
呼叫 藍牙客戶端1 .發送文字
文字 " S "


9:48
Screen1
Bluetooth List
連線
終止連線
連線狀態 NONE
鏟具上升
鏟具下降
水平布局1
BTList
水平布局2
Link
disconnect
水平布局3
標籤1
status
表格布局1
前
後
右
左
水平配置1
鏟具上
鏟具下
對話框1
重新命名
刪除
素材
DOWN.jpg
LEFT.jpg


當 右 .被壓下
執行 如果 藍牙客戶端1 . 連線狀態
則 呼叫 藍牙客戶端1 .發送文字
文字 " R "

當 右 .被鬆開
執行 如果 藍牙客戶端1 . 連線狀態
則 呼叫 藍牙客戶端1 .發送文字
文字 " S "


當 左 .被壓下
執行 如果 藍牙客戶端1 . 連線狀態
則 呼叫 藍牙客戶端1 .發送文字
文字 " L "


當 左 .被鬆開
執行 如果 藍牙客戶端1 . 連線狀態
則 呼叫 藍牙客戶端1 .發送文字
文字 " S "


9:48
Screen1
Bluetooth List
連線
終止連線
連線狀態 NONE
鏟具上升
鏟具下降
水平布局1
BTList
水平布局2
Link
disconnect
水平布局3
標籤1
status
表格布局1
前
後
右
左
水平配置1
鏟具上
鏟具下
對話框1
重新命名
刪除
素材
DOWN.jpg
LEFT.jpg


當 鏟具上 .被點選
執行 如果 藍牙客戶端1 . 連線狀態
則 呼叫 藍牙客戶端1 .發送文字
文字 " U "
當 鏟具下 .被點選
執行 如果 藍牙客戶端1 . 連線狀態
則 呼叫 藍牙客戶端1 .發送文字
文字 " D "