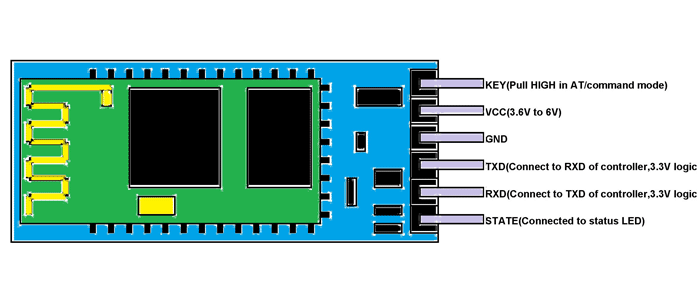
**Giao tiếp PIC – HC06**

**A, Sơ lược về Module Bluetooth HC06**

HC06 là một module Bluetooth được thiết kế để giao tiếp, truyền dữ liệu không dây tầm ngắn giữa 2 vi điều khiển hoặc hệ thống. Module này hoạt động trên giao thức giao tiếp Bluetooth 2.0 và nó chỉ có thể hoạt động ở chế độ “slave”.



|  |  |  |
| --- | --- | --- |
| **Pin** | **Ký hiệu** | **Chức năng** |
| 1 | Key/EN | Chân xác định trạng thái xem module hoạt động ở chế độ sử dụng lệnh AT hoặc chế độ bình thường |
| 2 | VCC | Chân cấp nguồn cho module |
| 3 | GND | Chân kết nối với đất |
| 4 | TXD | Dữ liệu nối tiếp được truyền theo module thông qua chân này, mức logic 3,3V |
| 5 | RXD | Dữ liệu nối tiếp được nhận bởi module thông qua chân này, mức logic 3,3V |
| 6 | State | Chân được kết nối với đèn led trên board để thể hiện trạng thái module |

**Các ưu điểm của HC06:**

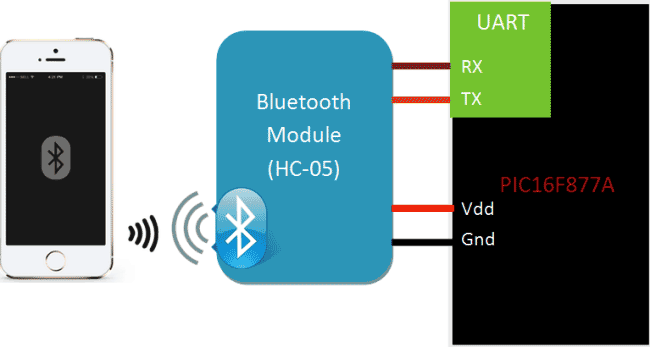
* Là lựa chọn tốt nhất khi cần giao tiếp không dây trong khoảng cách ngắn (dưới 100m).
* Dễ dàng giao tiếp và kết nối
* Là 1 trong những giải phát giá rẻ nhất cho giao tiếp không dây
* Tiêu thụ rất ít năng lượng để hoạt động và có thể được sử dụng trên các hệ thống di động chạy bằng pin
* Module HC06 có thể giao tiếp được với hầu hết các bộ điều khiển hoặc bộ xử lý như giao tiếp UART

**B, Giao tiếp giữ PIC16F877A với Module HC06**

**I, Cách kết nối chân**

Trong bài viết này, chúng ta sẽ bật/ tắt đèn led bằng cách gửi lệnh từ smart phone thông qua ứng dụng **Bluetooth Terminal**. Nếu chúng ta gửi ký tự ‘1’ từ ứng dụng, led sẽ bật, nếu gửi ‘0’ led sẽ tắt.

Sơ đồ khối sẽ được setup như hình dưới đây:



Trong 6 chân của Module HC06, chúng ta sẽ sử dụng 4 chân và được kết nối với MCU như sau:

|  |  |
| --- | --- |
| Chân trên HC06 | Chân trên MCU |
| VCC | VDD |
| GND | GND |
| TX | RC7/RX |
| RX | RC6/TX |

**II, Lập trình MCU PIC16F877A giao tiếp với Module HC06**

Cách lập trình để giao tiếp giữa PIC16F877A với Module HC06 khá giống với giao thức UART. Vì vậy bạn nên đọc trước bài viết về giao thức UART trong PIC16F877A.

**1, Khởi tạo giao tiếp Bluetooth cho PIC16F877A**

void BlueTooth\_Init**(**void**)**

**{**

//config pins for RX, TX

TRISC**.**B6 **=** 0**;** //TX pin set as output

TRISC**.**B7 **=** 1**;** //RX pin set as input

//initialize SPBRB register for baudrate

BRGH\_bit **=** 1**;** //high baud rate

SPBRG **=** 51**;** //for baudrate = 9600

//enable UART mode

SYNC\_bit **=** 0**;** //Asynchronous mode

SPEN\_bit **=** 1**;** //serial port enable

//config for transmission and reception

TXEN\_bit **=** 1**;** //transmit enable

CREN\_bit **=** 1**;** //enable continous receive

TX9\_bit **=** 0**;** //8 bits transmission

RX9\_bit **=** 0**;** //8 bits reception

//enable interrupt for RX TX

GIE\_bit **=** 1**;** //enable global interrupt

PEIE\_bit **=** 1**;** //enable peripharal interrupt

RCIE\_bit **=** 1**;** //enable RX interrupt

TXIE\_bit **=** 1**;** //enable TX interrupt

**}**

Hầu hết các module Bluetooth làm việc với baud rate 9600, module HC06 không là ngoại lệ. Vì thế trong bài này chúng ta sẽ điền giá trị cho thanh ghi **SPBRG** là 51(thạch anh được sử dụng ở đây là 8MHz) và cấu hình cho bit **BRGH** ở mức cao để chọn chế độ High Speed.

**2, Loading dữ liệu thông qua Bluetooth**

Không giống như UART, chúng ta cần xem xét vài điều ở đây trước khi có thể truyền và nhận dữ liệu. Module Bluetooth có bộ đệm truyền và nhận bên trong nó, dữ liệu được gửi tới nó sẽ được lưu trữ trong bộ RX Buffer. Dữ liệu này sẽ không được phát (gửi trên không) trừ khi “carriage return” được gửi tới module. Do đó để gửi dữ liệu chúng ta cần phải load vào RX Buffer của HC06 sau đó phát nó sử dụng “carriage return”.

Các công việc trên có thể dễ dàng thực hiện bằng các sử dụng các hàm chức năng. Hàm dưới đây được sử dụng khi chúng ta muốn load 1 ký tự vào bộ đệm RX. Chúng ta load dữ liệu vào thanh ghi **TXREG** và đợi cho đến khi nó được xử lý bằng cách kiểm tra cờ **TXIF** và **TRMT** bằng cách sử dụng các vòng while.

void BlueTooth\_SendByte**(**char myData**)**

**{**

TXREG **=** myData**;** //load the data to TX buffer

**while(!**TXIF\_bit**);** //wait complete transmitting

**while(!**TRMT\_bit**);**

**}**

Hàm tiếp theo được sử dụng để gửi 1 chuỗi vào bộ đệm RX của Module HC06. Chuỗi sẽ được chia nhỏ thành các ký tự và gửi các ký tự đi.

void BlueTooth\_Write**(**char **\***str**)**

**{**

**while(\***str**)** //if character != NULL

**{**

BlueTooth\_SendByte**(\***str**++);**

**}**

**}**

Đến giờ, chúng ta mới truyền thông tin vào bộ đệm RX của Module HC06. Bây giờ ta cần phải giúp nó phát dữ liệu qua không trung bằng cách sử dụng hàm dưới đây:

void BlueTooth\_BroadCast**(**void**)**

**{**

TXREG **=** 13**;**

Delay\_ms**(**300**);**

**}**

Trong hàm này, chúng ta sẽ gán giá trị 13 vào thanh ghi **TXREG**. Giá trị 13 này chỉ là mã ASCII để sử dụng cho “carriage return”.

**3, Nhận dữ liệu từ Bluetooth**

Tương tự như UART, hàm dưới đây được sử dụng để nhận dữ liệu từ Bluetooth.

char BlueTooth\_GetChar**(**void**)**

**{**

**if(**OERR\_bit**)** // check for over run error

**{**

CREN\_bit **=** 0**;**

CREN\_bit **=** 1**;** //Reset CREN

**}**

**if(**RCIF\_bit **==** 1**)** //if the user has sent a char return the char (ASCII value)

**{**

**while(!**RCIF\_bit**);**

**return** RCREG**;**

**}**

**else** //if user has sent no message return 0

**return** 0**;**

**}**

Nếu người dùng đã gửi dữ liệu, hàm này sẽ trả về dữ liệu cụ thể, có thể được lưu trong một biến và được xử lý. Nếu người dùng chưa gửi bất cứ thứ gì, hàm sẽ trả về 0.

**4, Chương trình chính**

Chúng ta sẽ sử dụng chân PB3 để kết nối với đèn led.

#include "blueTooth.h"

void main**()**

**{**

int get\_value **=** 0 **;**

TRISB3\_bit **=** 0**;** //PB3 set as output

PORTB**.**B3 **=** 0**;**

BlueTooth\_Init**();**

//Send Introductory

BlueTooth\_Write**(**"Bluetooth Initialized and Ready"**);**

BlueTooth\_BroadCast**();**

BlueTooth\_Write**(**"Press 1 to turn on LED"**);**

BlueTooth\_BroadCast**();**

BlueTooth\_Write**(**"Press 0 to turn on LED"**);**

BlueTooth\_BroadCast**();**

**while(**1**)**

**{**

get\_value **=** BlueTooth\_GetChar**();** //read the char from BT

**if(**get\_value **==** 0**)**

**{**

PORTB**.**B3 **=** 0**;**

BlueTooth\_Write**(**"Led OFF"**);**

BlueTooth\_BroadCast**();**

**}**

**if(**get\_value **==** 1**)**

**{**

PORTB**.**B3 **=** 1**;**

BlueTooth\_Write**(**"Led ON"**);**

BlueTooth\_BroadCast**();**

**}**

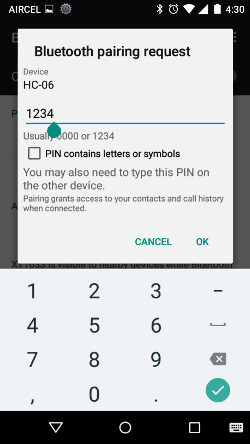
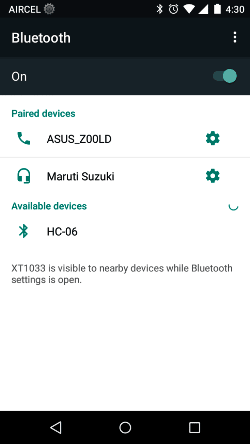
**}**

**}**

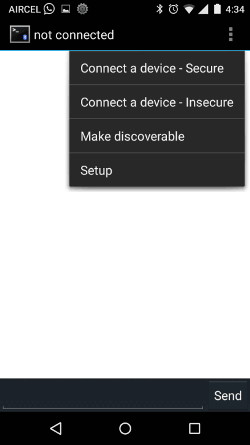
**III, Sử dụng ứng dụng để điều khiển Led**

Chúng ta cần cài đặt ứng dụng **Bluetooth Terminel** để kết nối với Module HC06, qua đó gửi dữ liệu từ app về cho MCU.

Trước hết, bạn phải kết nối smart phone với Module HC06, mật khẩu thường là “1234” hoặc “0000”



Sau khi smart phone kết nối thành công với HC06, chúng ta mở app ra và lựa chọn HC06 để kết nối với ứng dụng



Bây giờ, chúng ta chỉ cần gửi ký tự ‘0’ hoặc ‘1’ để điều khiển led:

