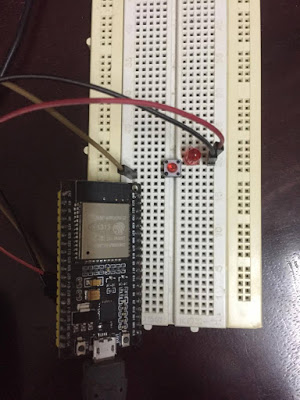
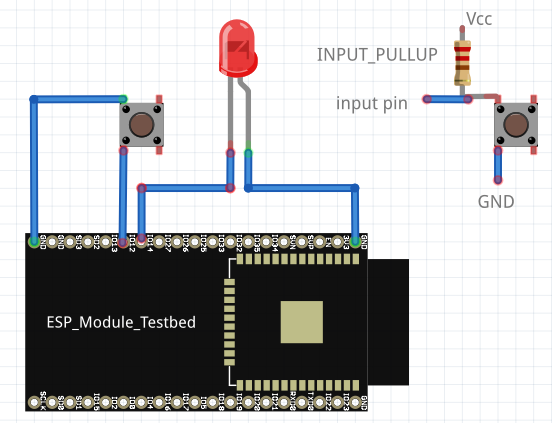
# **[Demo 21: How to use interrupt in Arduino ESP32](http://www.iotsharing.com/2017/06/how-to-use-interrupt-in-arduino-esp32.html)**

**1. Introduction**  
- When interrupt occurs, micro-controller will stop current task immediately to process the event that cause the interrupt. If we do not use interrupt, we have to use loop control statement to always listen on events that we are interested in. In case our program has many tasks to do, we may miss the interested event because micro-controller is busy to process another task.  
- The demo for this tutorial is quite simple. We use a button to connect to an interrupt pin. Every time the user press the button, it will cause an interrupt on ESP32 and ESP32 process this interrupt by toggle a LED.  
**2. Hardware**

[聽](https://2.bp.blogspot.com/-pfa1EC7u9zI/WUKtiPhhviI/AAAAAAAAED8/efce35pRP68oGYdXin030s0qD87ja24fACLcBGAs/s1600/esp32interrupt.jpg)

**Figure: ESP32 attach external interrupt**

Here we set interrupt pin as INPUT\_PULLUP, it means there is a resister connect the input pin to Vcc so our circuit is more simple.

[](https://4.bp.blogspot.com/-SAIoFz75OyQ/WUKx-FTo5eI/AAAAAAAAEEA/-ofagWKd738c4QmuCLYLtte5AtVFEplqACLcBGAs/s1600/esp32interrupt2.png)

**Figure: ESP32 connect to button (input\_pullup) and LED**

Connections:

[ESP32 GIO12 - BUTTON - GND]

[ESP32 GIO14 - LED - GND] **3. Software**  
- In order to use interrupt, we will use the 2 functions:聽**attachInterrupt(digitalPinToInterrupt(pin), ISR, mode)**  
+ pin: is the pin to attach interrupt.  
+ ISR: the function to be invoked when interrupt occurs.  
+ mode: which event trigger interrupt (**CHANGE**: trigger when the pin change value,聽**RISING**: when the pin go from LOW to HIGH, ...)  
- Create an Arduino project with code:

|  |
| --- |
| /\* LED pin \*/  byte ledPin = 14;  /\* pin that is attached to interrupt \*/  byte interruptPin = 12;  /\* hold the state of LED when toggling \*/  volatile byte state = LOW;  void setup() {  pinMode(ledPin, OUTPUT);  /\* set the interrupt pin as input pullup\*/  pinMode(interruptPin, INPUT\_PULLUP);  /\* attach interrupt to the pin  function blink will be invoked when interrupt occurs  interrupt occurs whenever the pin change value \*/  attachInterrupt(digitalPinToInterrupt(interruptPin), blink, CHANGE);  }  void loop() {  }  /\* interrupt function toggle the LED \*/  void blink() {  state = !state;  digitalWrite(ledPin, state);  } |