

## Manuals and Curriculum

(<http://playground.arduino.cc/Main/ManualsAndCurriculum>)

## Arduino StackExchange

(<http://arduino.stackexchange.com>)

## Board Setup and Configuration

(<http://playground.arduino.cc/Main/ArduinoBoardSetup>)

## Development Tools

(<http://playground.arduino.cc/Main/DevelopmentTools>)

## Arduino on other Atmel Chips

(<http://playground.arduino.cc/Main/ArduinoOnOtherAtmelChips>)

## Interfacing With Hardware

(<http://playground.arduino.cc/Main/InterfacingWithHardware>)

## - Output

(<http://playground.arduino.cc/Main/InterfacingWithHardware#Output>)

## - Input

(<http://playground.arduino.cc/Main/InterfacingWithHardware#Input>)

## - User Interface

(<http://playground.arduino.cc/Main/InterfacingWithHardware#UI>)

## - Storage

(<http://playground.arduino.cc/Main/InterfacingWithHardware#Storage>)

## - Communication

(<http://playground.arduino.cc/Main/InterfacingWithHardware#Communication>)

## - Power supplies

(<http://playground.arduino.cc/Main/IntWithHW-PwrSup>)

## - General

(<http://playground.arduino.cc/Main/InterfacingWithHardware#General>)

## Interfacing with Software

(<http://playground.arduino.cc/Main/InterfacingWithSoftware>)

## User Code Library

(<http://playground.arduino.cc/Main/GeneralCodeLibrary>)

## - Snippets and Sketches

(<http://playground.arduino.cc/Main/SketchList>)

## - Libraries

(<http://playground.arduino.cc/Main/LibraryList>)

## - Tutorials

(<http://playground.arduino.cc/Main/TutorialList>)

## Suggestions &amp; Bugs

(<http://code.google.com/p/arduino/issues/list>)

## Electronics Technique

(<http://playground.arduino.cc/Main/ElectroInfoResources>)

## Sources for Electronic Parts

(<http://playground.arduino.cc/Main/Resources>)

## Related Hardware and Initiatives

(<http://playground.arduino.cc/Main/SimilarBoards>)

## Arduino People/Groups &amp; Sites

(<http://playground.arduino.cc/Main/People>)

## Exhibition

(<http://playground.arduino.cc/Projects/ArduinoHsts>)

## Project Ideas

(<http://playground.arduino.cc/Projects/Ideas>)

## Languages

(<http://playground.arduino.cc/Main/Languages>)

## Interrupts

The processor at the heart of any Arduino has two different kinds of interrupts: “external”, and “pin change”. There are only two external interrupt pins on the ATmega168/328 (ie, in the Arduino Uno, Nano, Duemilanove), INT0 and INT1, and they are mapped to Arduino pins 2 and 3. These interrupts can be set to trigger on RISING or FALLING signal edges, or on low level. The triggers are interpreted by hardware, and the interrupt is very fast. The Arduino Mega has a few more external interrupt pins available.

On the other hand the pin change interrupts can be enabled on many more pins. For ATmega168/328-based Arduinos, they can be enabled on any or all 20 of the Arduino's signal pins; on the ATmega-based Arduinos they can be enabled on 24 pins. They are triggered equally on RISING or FALLING signal edges, so it is up to the interrupt code to set the proper pins to receive interrupts, to determine what happened (which pin? ...did the signal rise, or fall?), and to handle it properly. Furthermore, the pin change interrupts are grouped into 3 “port”s on the MCU, so there are only 3 interrupt vectors (subroutines) for the entire body of pins. This makes the job of resolving the action on a single interrupt even more complicated.

## Simple External Interrupt examples

- Handling external interrupts with Arduino (<http://gonium.net/md/2006/12/20/handling-external-interrupts-with-arduino/>)

With Arduino 007, there is a simpler method for this: see `attachInterrupt(..)`

(<http://www.arduino.cc/en/Reference/AttachInterrupt>)

- Handling timer interrupts (<http://gonium.net/md/2006/12/27/i-will-think-before-i-code/>)

## More complex examples

- BCF77 receiver clock (<http://gonium.net/md/2007/04/18/tweaking-the-code/>) uses both external and timer interrupts.

- Interfacing with the Parallax ultrasonic PING module using interrupts - Code (<http://playground.arduino.cc/Code/PingInterruptCode>)

## Pin Change Interrupt Library and Examples

- See <http://code.google.com/p/arduino-pinchangeint/> (<http://code.google.com/p/arduino-pinchangeint/>) for an example of a library that implements Pin Change Interrupts. Example code is included.

An object-oriented version of the library is (...will be, as of 2/3/2012) available at

<http://code.google.com/p/oopinchangeint/> (<http://code.google.com/p/oopinchangeint/>)

- PciManager (<http://code.google.com/p/arduino-pcimanager/>) is an other implementation for

Partecipate  
(<http://playground.arduino.cc/Main/Participate>)

- Formatting guidelines  
(<http://playground.arduino.cc/Main/Participate#contribrules>)
- All recent changes  
(<http://playground.arduino.cc/Site/AllRecentChanges>)
- PmWiki  
(<http://playground.arduino.cc/PmWiki/PmWiki>)
- WikiSandBox training  
(<http://playground.arduino.cc/Main/WikiSandbox>)
- Basic Editing  
(<http://playground.arduino.cc/PmWiki/BasicEditing>)
- Documentation index  
(<http://www.pmwiki.org/wiki/PmWiki/DocumentationIndex>)

## Share



helping developers to use PCI. You can register more listeners on more pins that should be launched on pin change.

- `attachInterrupt()` (<http://arduino.cc/en/Reference/attachInterrupt>)
- `interrupts()` (<http://arduino.cc/en/Reference/Interrupts>)
- `PinChangeInt` (<http://playground.arduino.cc/Main/PinChangeInt>)

## NEWSLETTER

Enter your email to sign up



©2015 Arduino [Copyright Notice](http://arduino.cc/en/Main/CopyrightNotice) (<http://arduino.cc/en/Main/CopyrightNotice>) [Contact us](http://arduino.cc/en/Main/ContactUs) (<http://arduino.cc/en/Main/ContactUs>)

<https://twitter.com/arduino>

<http://www.facebook.com/official.arduino>

<https://plus.google.com/+Arduino>

[http://www.flickr.com/photos/arduino\\_cc](http://www.flickr.com/photos/arduino_cc)

<http://youtube.com/arduinoteam>