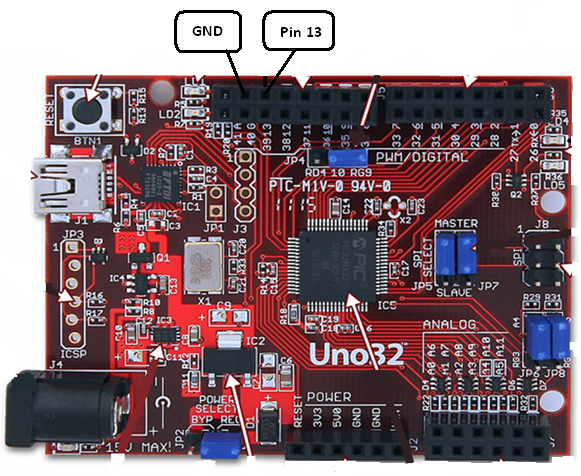
**BLINK WITHOUT DELAY**

Sometimes you need to do two things at once. For example you might want to blink an LED (or some other time-sensitive function) while reading a button press or other input. In this case, you can't use delay(), or you'd stop everything else the program while the LED blinked. The program might miss the button press if it happens during the delay(). This sketch demonstrates how to blink the LED without using delay(). It keeps track of the last time the Arduino turned the LED on or off. Then, each time through loop(), it checks if a long enough interval has passed. If it has, it toggles the LED on or off.

**Hardware Required:**

* Arduino Board
* LED

**Circuit:**

****

To build the circuit, grab an LED and attach its long, positive leg (called the anode) to pin 13. Attach the short, negative leg (called the anode) to ground. Then plug your Arduino board into your computer, start the Arduino program, and enter the code below.

**Code:**

The code below uses the [millis()](http://arduino.cc/en/Reference/Millis) function, a command that returns the number of milliseconds since the Arduino board started running its current program, to blink an LED.

*/\* Blink without Delay  
   
 Turns on and off a light emitting diode(LED) connected to a digital    
 pin, without using the delay() function.  This means that other code  
 can run at the same time without being interrupted by the LED code.  
   
 The circuit:  
 \* LED attached from pin 13 to ground.  
 \* Note: on most Arduinos, there is already an LED on the board  
 that's attached to pin 13, so no hardware is needed for this example.  
 \*/*  
  
*// constants won't change. Used here to*  
*// set pin numbers:*  
const int ledPin =  13;      *// the number of the LED pin*  
  
*// Variables will change:*  
int ledState = LOW;             *// ledState used to set the LED*  
long previousMillis = 0;        *// will store last time LED was updated*  
  
*// the follow variables is a long because the time, measured in miliseconds,*  
*// will quickly become a bigger number than can be stored in an int.*  
long interval = 1000;           *// interval at which to blink (milliseconds)*  
  
void **setup**() {  
  *// set the digital pin as output:*  
  pinMode(ledPin, OUTPUT);        
}  
  
void **loop**()  
{  
  *// here is where you'd put code that needs to be running all the time.*  
  
  *// check to see if it's time to blink the LED; that is, if the*  
  *// difference between the current time and last time you blinked*  
  *// the LED is bigger than the interval at which you want to*  
  *// blink the LED.*  
  unsigned long currentMillis = millis();  
   
  if(currentMillis - previousMillis > interval) {  
    *// save the last time you blinked the LED*  
    previousMillis = currentMillis;     
  
    *// if the LED is off turn it on and vice-versa:*  
    if (ledState == LOW)  
      ledState = HIGH;  
    else  
      ledState = LOW;  
  
    *// set the LED with the ledState of the variable:*  
    digitalWrite(ledPin, ledState);  
  }  
}