Arduino as Microcontroller ISP

Embedded Workshop 8/26/2015

Arduino as ISP Programmer

Arduino as In-System Programmer

1. Connect Arduino to computer and validate serial Port

   Tools > Port

2. Select the Arduino Uno board:

   Tools > Board > Arduino Uno

3. Load the sketch:

   File > Examples > ArduinoISP > Click upload

4. Change Programmer:

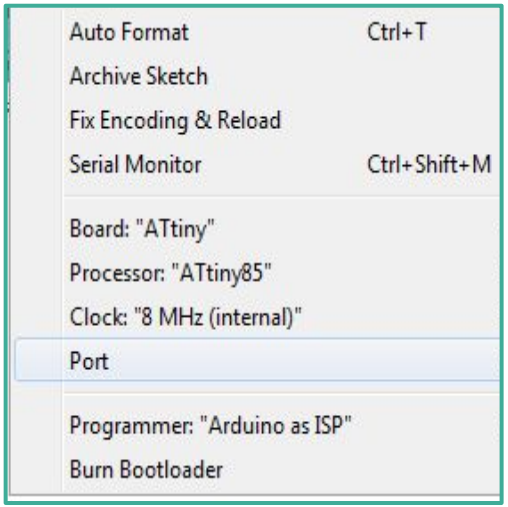
   Tools > Programmer > Arduino as ISP

5. Disconnect USB cable

6. Connect Arduino to ATtiny85 or ATmega328 using jumper wires and breadboard

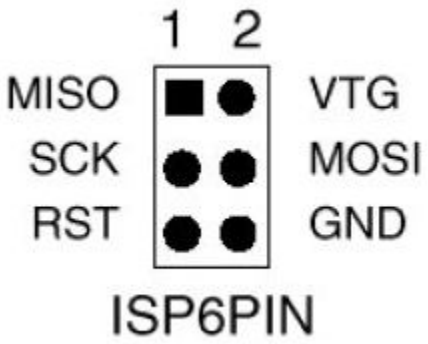
Be sure to set Programmer to:

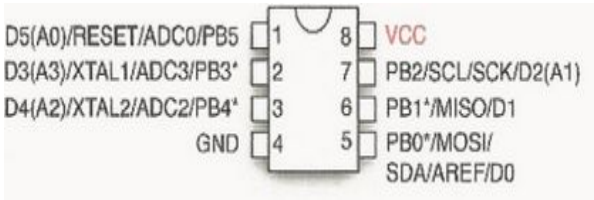
"**Arduino as ISP**"

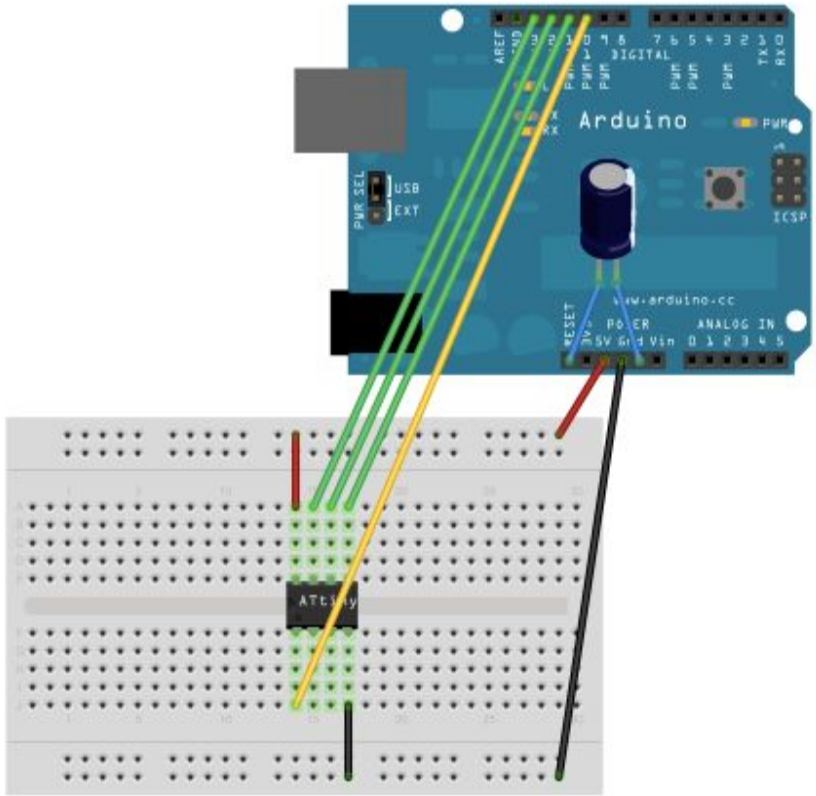


Connect Arduino to ATtiny 85

1. ATtiny **Pin 7(PB2)** to Arduino **Pin 13** (or SCK of another programmer)
2. ATtiny **Pin 6(PB1)** to Arduino **Pin 12** (or MISO of another programmer)
3. ATtiny **Pin 5(PB0)** to Arduino **Pin 11** (or MOSI of another programmer)
4. ATtiny Reset **Pin 1 (PB5)** to Arduino **Pin 10** (or RESET of another programm
5. Connect a **10uF** capacitor between **reset** and **ground** on the Arduino board
6. Add LED **Pin 5(PB0)** and connect 330 ohm resistor to LED and GND
7. Connect 5volt **Pin 8(VCC)**
8. Gnd **Pin 4(GND)**

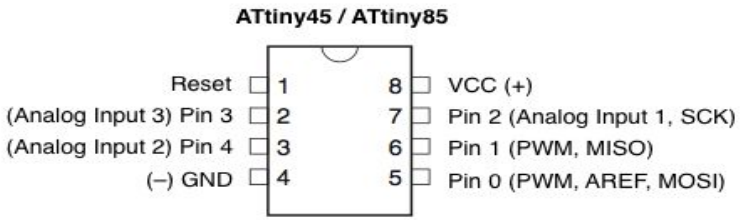






Installing the ATtiny support using the built-in board's manager

1. Go to Highlowtech New version1.6 http://highlowtech.org/?p=1695
2. ***File > Preferences***. Find the "Additional Boards Manager URLs" field near the bottom of the dialog. Paste the following URL into the field:
3. https://raw.githubusercontent.com/damellis/attiny/ide-1.6.x-boards-manager/package_damellis_attiny_index.json
4. Click the OK button to save your updated preferences.
5. Open the board's manager in the ***Tools > Board > Board Manager***.
6. Scroll to the bottom of the list; you should see an entry for "***ATtiny***".
7. Click on the ***ATtiny*** entry. An install button should appear. Click the ***install*** button.
8. The word "installed" should now appear next to the title of the ATtiny entry.
9. Close the board's manager. You should now see an entry for ATtiny in the "***Tools > Board***" menu.
10. Select ***ATtiny85*** as your board. Select the following also:
    a. ***Tools > Processors > ATtiny85***
    b. ***Tools > Clock > 8 mHz internal***



Flash ATtiny85 with the Blink Sketch

1. Plug USB cable to Arduino Uno with attached breadboard with 85 chip.
2. Setup Arduino IDE with blink Sketch: ***File > Examples > Basics > Blink*** (***Change pin 13 to 0***)
3. Double check your IDE settings:
   a. ***Tools > Board > ATtiny85***
   b. ***Tools > Processors > ATtiny85***
   c. ***Tools > Clock > 8 mHz internal***
   d. ***Tools > Programmer > Arduino as ISP*** (not ArduinoISP)
4. Flash the Sketch to the ATtiny85 chip: ***File > Upload using programmer***
5. After Flash completes the LED should be blinking

If this does not work try the following:

Recheck all settings and connections

Try using: Burn Bootloader.

Try another ATtiny.

```
void setup() {
  // initialize digital pin 13 as an output.
  pinMode(0, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(0, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(500);              // wait for a second
  digitalWrite(0, LOW);    // turn the LED off by making the voltage LOW
  delay(500);              // wait for a second
}
```

ATmega 328 Pinout

Use the pinout for connecting the ATmega328 to the Breadboard.

Do not need to change the Pin 13 to Pin 0 in the sketch.

Select ATmega as your board.