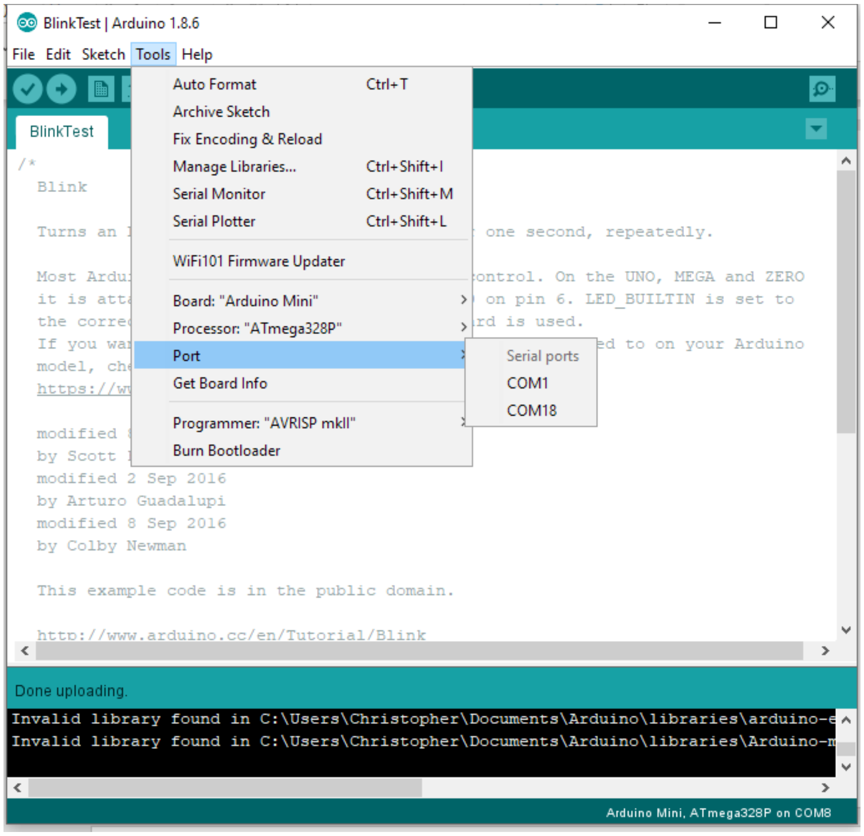
Uploading First Program to ATMEGA Controller

Procedure for Uploading Blink Test

Step 1: Open your Arduino IDE and check the COM Ports attached to computer by clicking on Tools and hovering over Ports. Take note of what is already in ports so when you check again after connecting your microcontroller, you can notice which port is new.



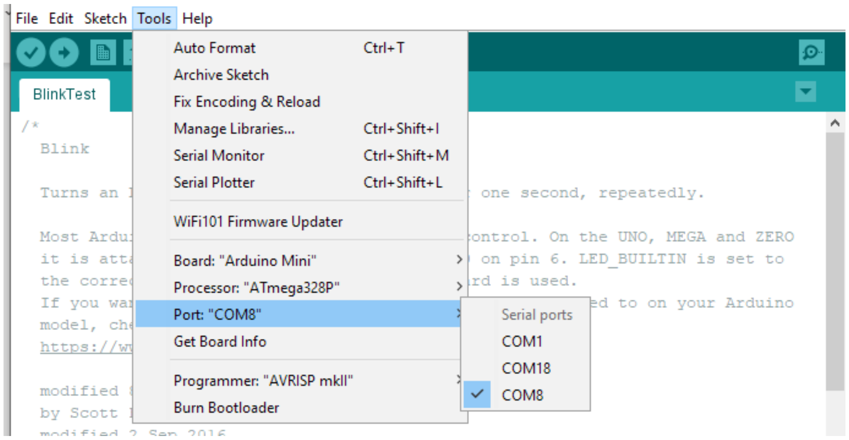
```
/*
  Blink

  Turns an I                                        one second, repeatedly.

  Most Ardu                                        ontrol. On the UNO, MEGA and ZERO
  it is att                                         on pin 6. LED_BUILTIN is set to
  the corre                                        rd is used.
  If you wa                                         ed to on your Arduino
  model, ch
  https://w

  modified
  by Scott
  modified 2 Sep 2016
  by Arturo Guadalupi
  modified 8 Sep 2016
  by Colby Newman

  This example code is in the public domain.

  http://www.arduino.cc/en/Tutorial/Blink
```

Done uploading.

```
Invalid library found in C:\Users\Christopher\Documents\Arduino\libraries\arduino-e
Invalid library found in C:\Users\Christopher\Documents\Arduino\libraries\Arduino-m
```

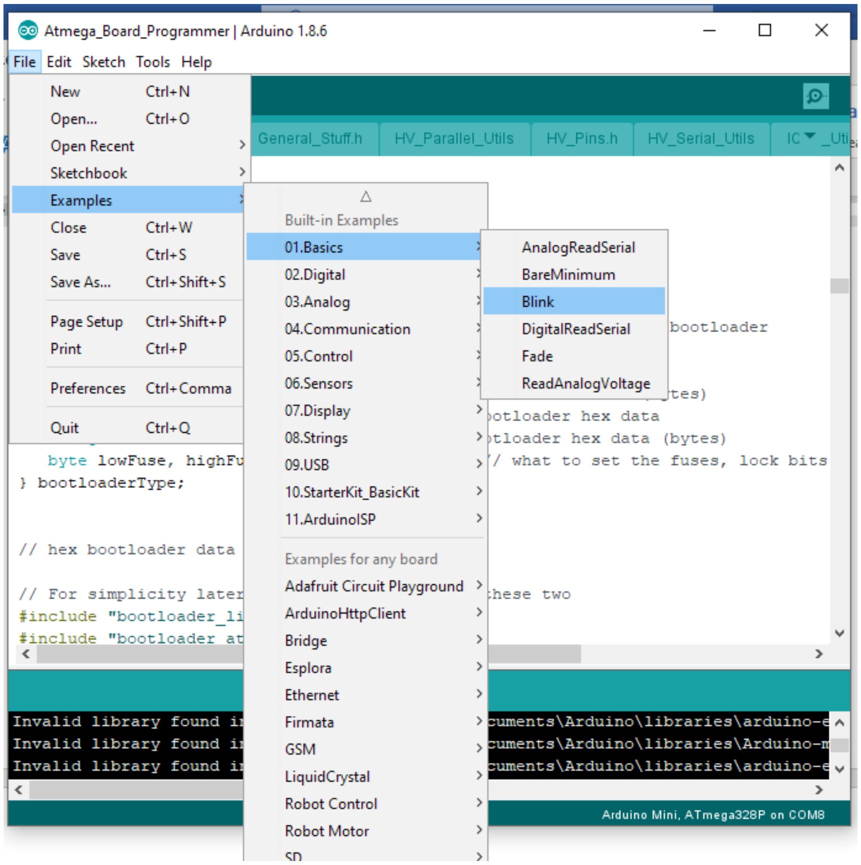
Arduino Mini, ATmega328P on COM8

Step 2: Connect USB from your PC to your new Microcontroller device. The computer should automatically create a new COM Port for your microcontroller. Check to see what port has now appeared. In my case it was Port 8.



```
/*
  Blink

  Turns an I                                        one second, repeatedly.

  Most Ardu                                        ontrol. On the UNO, MEGA and ZERO
  it is att                                         on pin 6. LED_BUILTIN is set to
  the corre                                        rd is used.
  If you wa                                         ed to on your Arduino
  model, ch
  https://w

  modified
  by Scott
  modified 2 Sep 2016
```

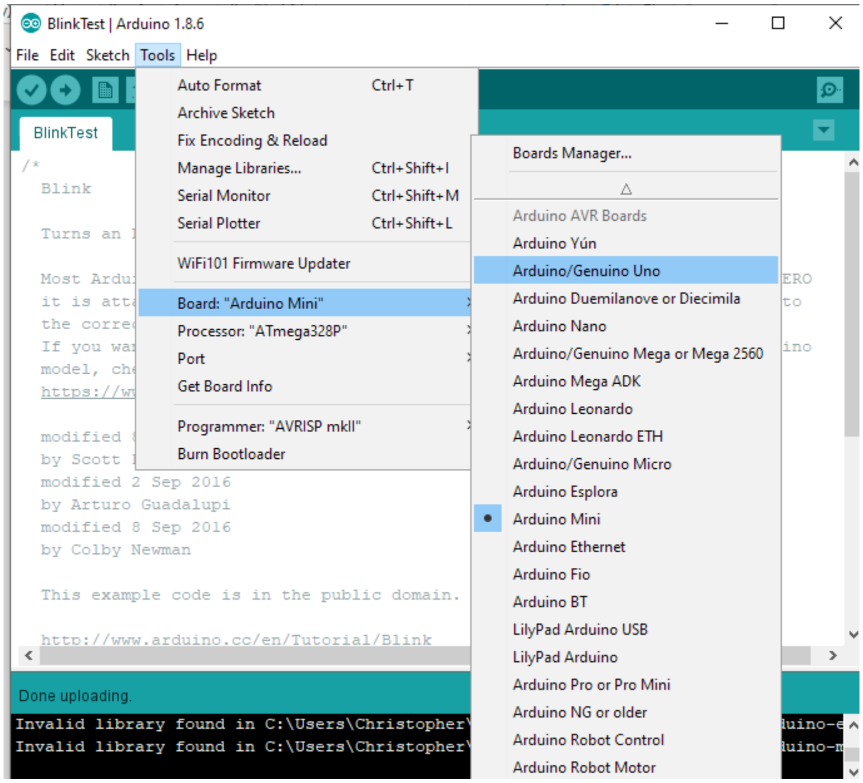
Step 3: Now find the test program called Blink to confirm the you can successfully upload to the ATMEGA controller. You can find this by clicking File followed by hovering over Examples and then Basics.



```
                                                bootloader

                                                tes)
                                            otloader hex data
                                            tloader hex data (bytes)
   byte lowFuse, highFu                     / what to set the fuses, lock bits
} bootloaderType;


// hex bootloader data

// For simplicity later                     hese two
#include "bootloader_li
#include "bootloader at
```

```
Invalid library found i                     cuments\Arduino\libraries\arduino-e
Invalid library found i                     cuments\Arduino\libraries\Arduino-m
Invalid library found i                     cuments\Arduino\libraries\arduino-e
```

Step 4: There are now a few features to double check before uploading the code. The first is the board type. Depending on the bootloader pre-installed on the microcontroller the board type could be either Arduino Mini or Arduino/Genuino Uno. If one doesn't end up working, try the other.

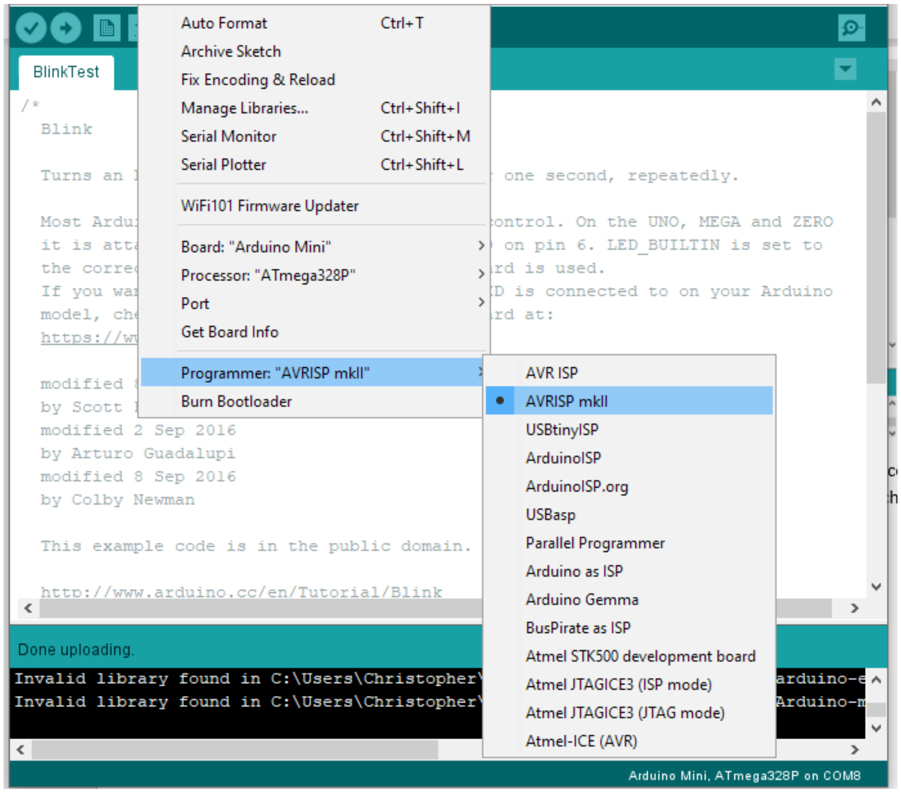


```
/*
  Blink

  Turns an 

  Most Ardu
  it is att
  the corre
  If you wa
  model, ch
  https://w

  modified 
  by Scott 
  modified 2 Sep 2016
  by Arturo Guadalupi
  modified 8 Sep 2016
  by Colby Newman

  This example code is in the public domain.

  http://www.arduino.cc/en/Tutorial/Blink
```

Done uploading.

```
Invalid library found in C:\Users\Christopher\
Invalid library found in C:\Users\Christopher\
```

Step 5: Double check the COM Port and make sure it is now on the COM for your Microcontroller as discovered in steps 1 & 2. Also, make sure that the Programmer says "AVRISP mkII" or change it also.



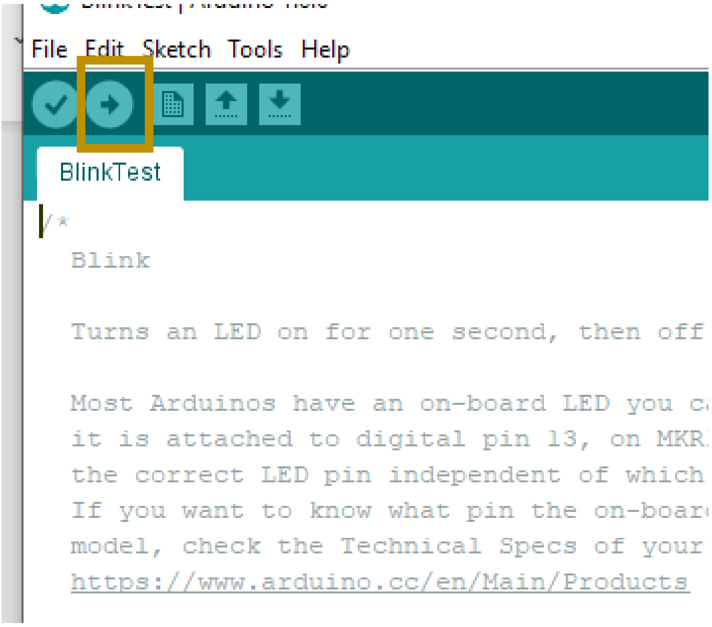
```
/*
  Blink

  Turns an                                    one second, repeatedly.

  Most Ardu                                  ontrol. On the UNO, MEGA and ZERO
  it is att                                   on pin 6. LED_BUILTIN is set to
  the corre                                  rd is used.
  If you wa                                  D is connected to on your Arduino
  model, ch                                  rd at:
  https://w

  modified 
  by Scott 
  modified 2 Sep 2016
  by Arturo Guadalupi
  modified 8 Sep 2016
  by Colby Newman

  This example code is in the public domain.

  http://www.arduino.cc/en/Tutorial/Blink
```

Done uploading.

```
Invalid library found in C:\Users\Christopher\                    arduino-e
Invalid library found in C:\Users\Christopher\                    Arduino-m
```

Arduino Mini, ATmega328P on COM8

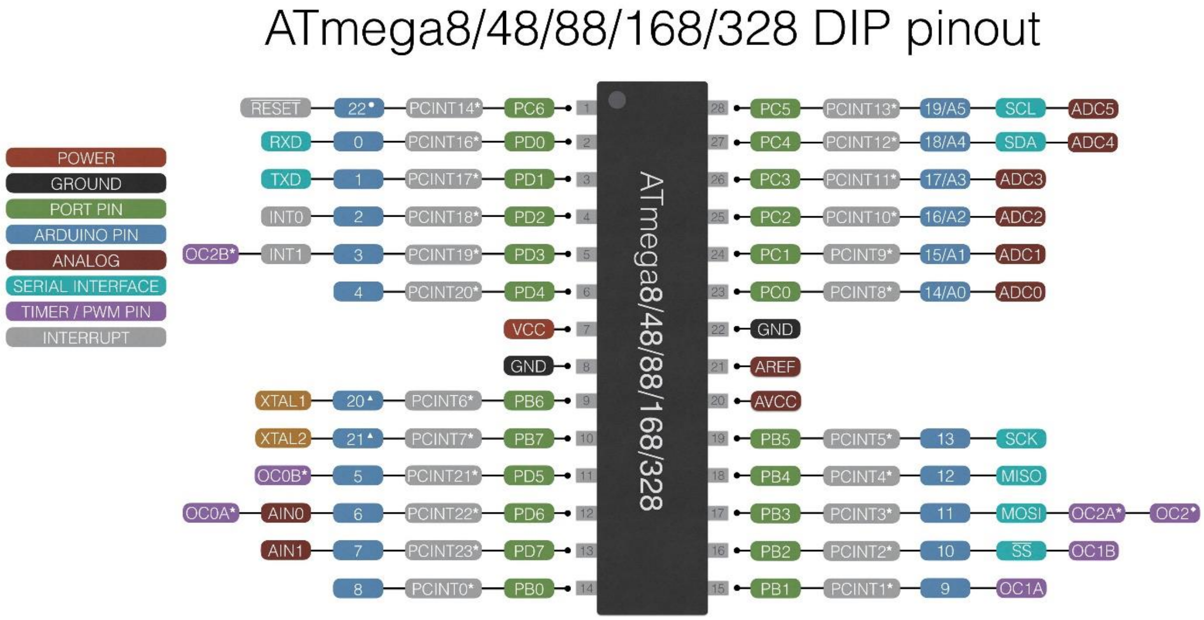
Step 6: Last but not least, upload the program by pressing the arrow button near the top left of the window. When the program is finished uploading it will say 'Upload Complete' in the lower right side of the window. If the program does not upload after a minute or two unplug and change the board type in Step 4, then try again. If the microcontroller does not blink, there maybe a wiring issue and you should review the schematic.



```
/*
  Blink

  Turns an LED on for one second, then off

  Most Arduinos have an on-board LED you c
  it is attached to digital pin 13, on MKR
  the correct LED pin independent of which
  If you want to know what pin the on-boar
  model, check the Technical Specs of your
  https://www.arduino.cc/en/Main/Products
```

Step 7: If you have not used this software before, save the program under a new name and try adjusting the values and test the results by uploading it to the microcontroller. Also, search the web for examples and follow some tutorials posted online. Many of them reference the Arduino Uno which uses the same microcontroller within the circuit you have just put together. Reference the pins shown below when following other tutorials.

ATMEGA Controller Pins



* ATmega48/88/168/328 only
• ATmega8 only
▲ Internal oscillator must be enabled
• Reset pin must be disabled

http://github.com/MCUdude/MiniCore

The ‘Arduino Pin’ numbers, in blue above, are the values used to represent each pin in the Arduino IDE code. Some of the pins have the letter A in the value which indicates that it is an analog pin. Analog pins can take a variable input voltage between 0 and 5 volts and produce a value that relates to the input voltage on that pin. The pins that also have an aqua colored label, have a serial interface capability. VCC and AVCC are for power and should have 5 volts going to each of them. GND represent the negative terminal of the power supply that powers the microcontroller.

Additional Information

If problems occur with the upload, double check that the procedure was followed, try both the Arduino Uno board option and the Arduino Mini board option, and if neither works then contact us through our website or email.