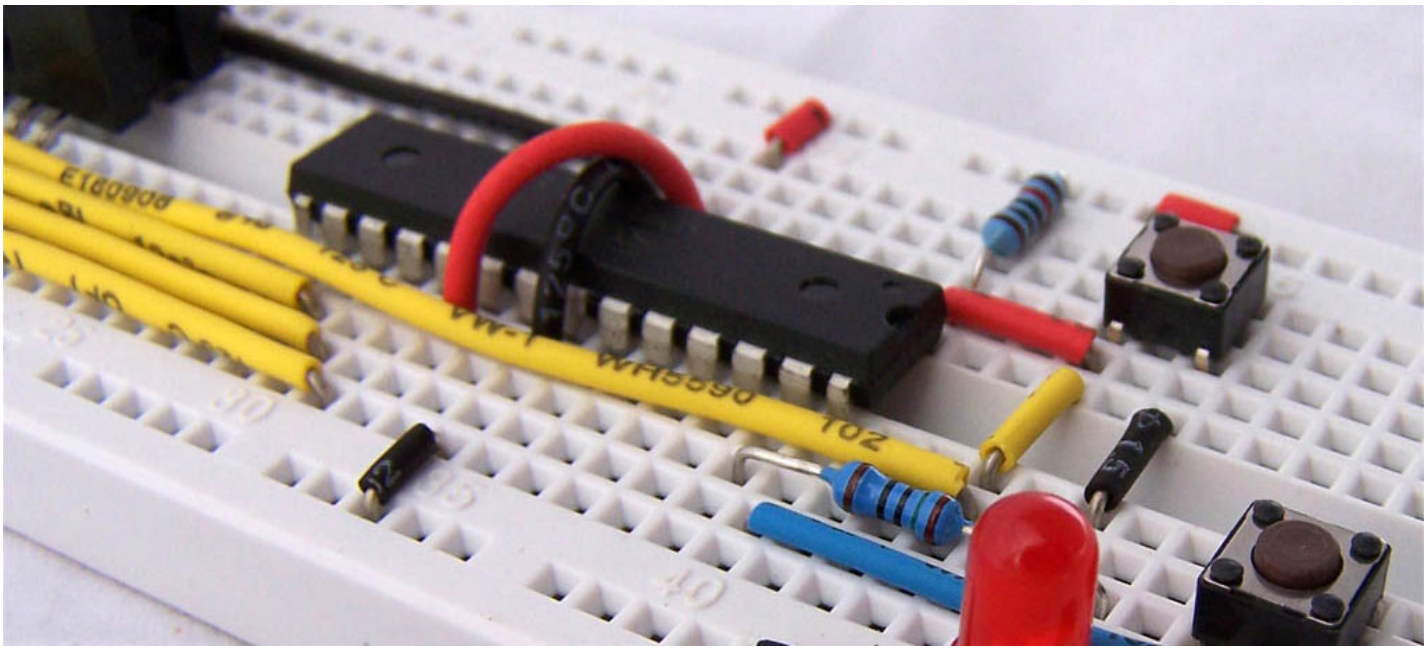


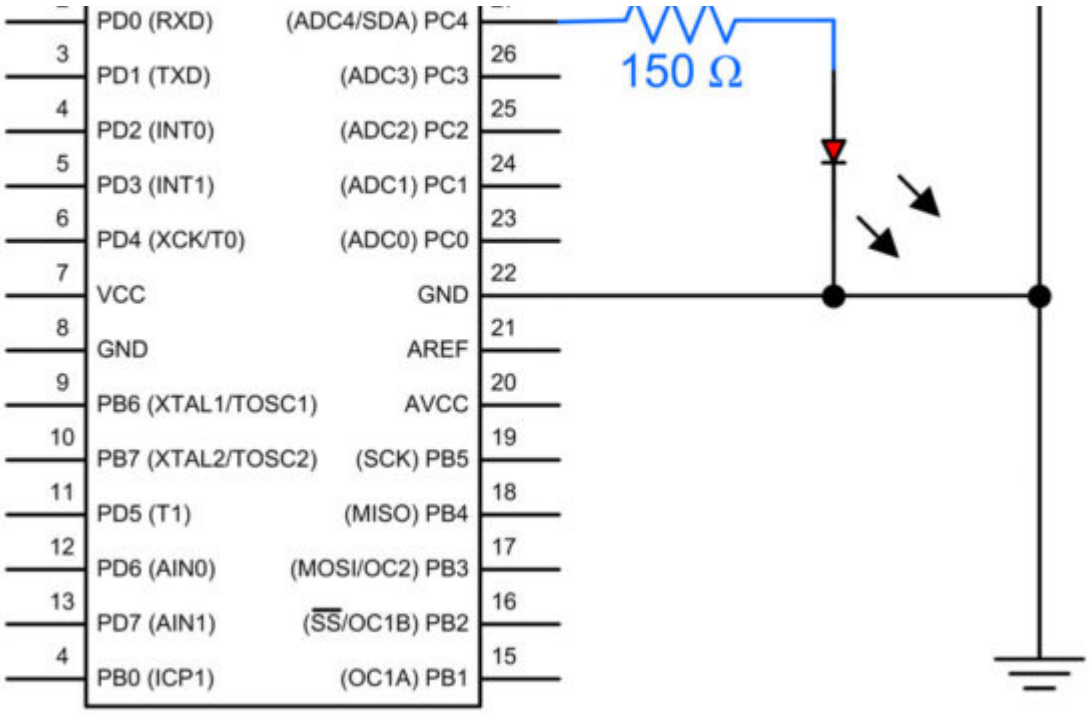
ATMEGA8 BREADBOARD CIRCUIT – PART 3 OF 3 – THE FIRMWARE

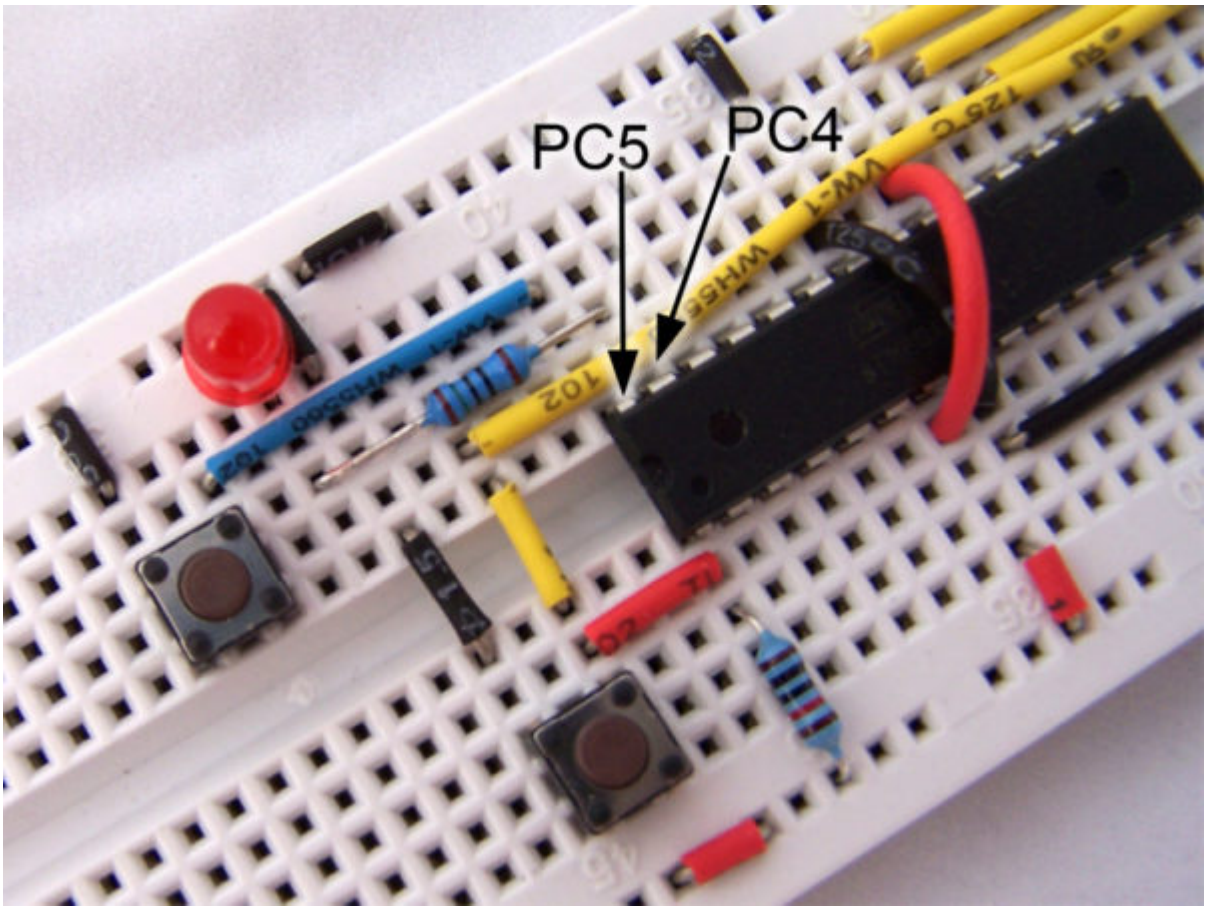
August 12, 2009 AVR Microcontrollers, Tutorials 25 Comments

This tutorial continues on from ATmega8 breadboard circuit Part 1 and ATmega8 breadboard circuit Part 2. So far we've built a power supply, added the microcontroller, added some plumbing to make it work and added the ISP interface, but it really doesn't do anything. The next step is to add some I/O devices and upload some firmware.

A lot of ATmega8 tutorials will use a "Hello World" program which consists of an LED that blinks at 1Hz. For this tutorial we will build on this and have an LED that blinks 3 times when a button is pressed. The first thing we need to do is add the LED and button to the breadboard.

The images below shows the schematic and the components on the breadboard.





The PC5 pin will be “pulled up” and when the button is pressed, this will ground the pin and initiate the blinking of the LED. The LED is connected to PC4 via a 150 ohm resistor. I calculated the resistor size using the LED Resistor Calculator on ohmslawcalculator.com.

the following options then save the file in a folder for this project.

- MCU Type: ATmega8
- Port: usb

Next we want to edit the makefile and change the following entries

```
AVRDUDE_PROGRAMMER = usbasp
```

```
F_CPU = 1000000
```

We need to modify the AVRDUDE_PROGRAMMER entry because the mfile utility does not have “usbasp” in the programmers list. This of course assumes that you are using a USBASP Programmer. If you are using another programmer, you will need to modify this section of the makefile with values that are suitable for that programmer.

The F_CPU entry specifies the microcontroller clock speed and is used by the _delay_ms function. The ATmega8 out of the box runs on a 1MHz internal clock, but this can be changed if required.

Using programmers notepad (Another tool bundled with WinAVR), create main.c with the following content.

```
#include <avr/io.h>
#include <util/delay.h>


//Define functions
//======================
void ioinit(void);
void led_on(void);
void led_off(void);
//======================

int main (void)
{
    ioinit(); //Setup IO pins and defaults

   while (1)
   {
      if (bit_is_clear(PINC, 5))
      {
         for (int i=0;i<3;i++)
         {
            if (i>0)
               _delay_ms(500);
            led_on();
            _delay_ms(500);
            led_off();
         }
      }
   }
}


```

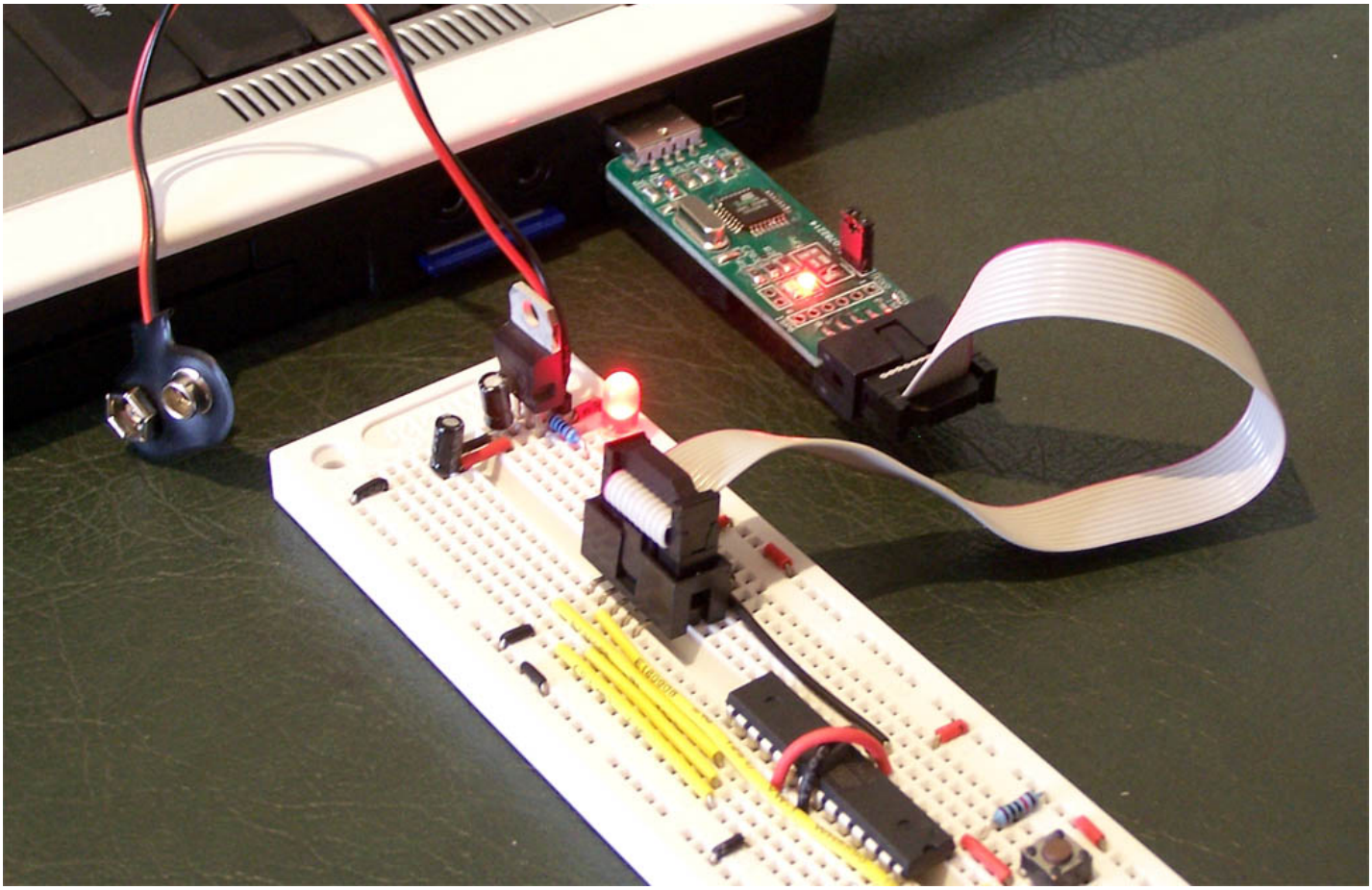
```
38
39 void led_on(void)
40 {
41    PORTC |= _BV(PC4);
42 }
43
44 void led_off(void)
45 {
46    PORTC &= ~_BV(PC4);
47 }
```

To compile the firmware

- Open a command prompt
- Make sure you are in the folder containing the makefile and main.c
- type “make” and press enter

```
Copyright (C) 2008 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.  There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.


Size before:
AVR Memory Usage
----------------
Device: atmega8

Program:     144 bytes (1.8% Full)
(.text + .data + .bootloader)

Data:          0 bytes (0.0% Full)
(.data + .bss + .noinit)



Compiling C: main.c
avr-gcc -c -mmcu=atmega8 -I. -gdwarf-2 -DF_CPU=1000000UL -Os -funsigned-char -funsigned-bitfields -f
pack-struct -fshort-enums -Wall -Wstrict-prototypes -Wa,-adhlns=./main.lst  -std=gnu99 -MMD -MP -MF
.dep/main.o.d main.c -o main.o

Linking: main.elf
avr-gcc -mmcu=atmega8 -I. -gdwarf-2 -DF_CPU=1000000UL -Os -funsigned-char -funsigned-bitfields -fpac
k-struct -fshort-enums -Wall -Wstrict-prototypes -Wa,-adhlns=main.o  -std=gnu99 -MMD -MP -MF .dep/ma
in.elf.d main.o --output main.elf -Wl,-Map=main.map,--cref      -lm

Creating load file for Flash: main.hex
avr-objcopy -O ihex -R .eeprom -R .fuse -R .lock main.elf main.hex

Creating load file for EEPROM: main.eep
avr-objcopy -j .eeprom --set-section-flags=.eeprom="alloc,load" \
        --change-section-lma .eeprom=0 --no-change-warnings -O ihex main.elf main.eep || exit 0

Creating Extended Listing: main.lss
avr-objdump -h -S -z main.elf > main.lss

Creating Symbol Table: main.sym
avr-nm -n main.elf > main.sym

Size after:
AVR Memory Usage
----------------
Device: atmega8

Program:     144 bytes (1.8% Full)
(.text + .data + .bootloader)

Data:          0 bytes (0.0% Full)
(.data + .bss + .noinit)



-------- end --------


C:\Users\dgarcia\Desktop\Blink3>_
```

To upload you firmware, first connect the USBASP Programmer to the USB port and breadboard circuit as shown in the photo below.



In the command prompt enter the following command then press enter

- make program

```
C:\Users\dgarcia\Desktop\Blink3>make program
avrdude -p atmega8 -P usb -c usbasp    -U flash:w:main.hex
avrdude: error: could not find USB device "USBasp" with vid=0x16c0 pid=0x5dc
make: *** [program] Error 1

C:\Users\dgarcia\Desktop\Blink3>make program
avrdude -p atmega8 -P usb -c usbasp    -U flash:w:main.hex

avrdude: AVR device initialized and ready to accept instructions

Reading | ################################################## | 100% 0.04s

avrdude: Device signature = 0x1e9307
avrdude: NOTE: FLASH memory has been specified, an erase cycle will be performed
         To disable this feature, specify the -D option.
avrdude: erasing chip
avrdude: reading input file "main.hex"
avrdude: input file main.hex auto detected as Intel Hex
avrdude: writing flash (144 bytes):

Writing | ################################################## | 100% 0.07s


avrdude: 144 bytes of flash written
avrdude: verifying flash memory against main.hex:
avrdude: load data flash data from input file main.hex:
avrdude: input file main.hex auto detected as Intel Hex
avrdude: input file main.hex contains 144 bytes
avrdude: reading on-chip flash data:

Reading | ################################################## | 100% 0.05s


avrdude: verifying ...
avrdude: 144 bytes of flash verified

avrdude: safemode: Fuses OK

avrdude done.  Thank you.

C:\Users\dgarcia\Desktop\Blink3>_
```

The system should now be ready for testing. Connect it up to your power source, press the button and watch the LED blink 3 times.

Related posts:

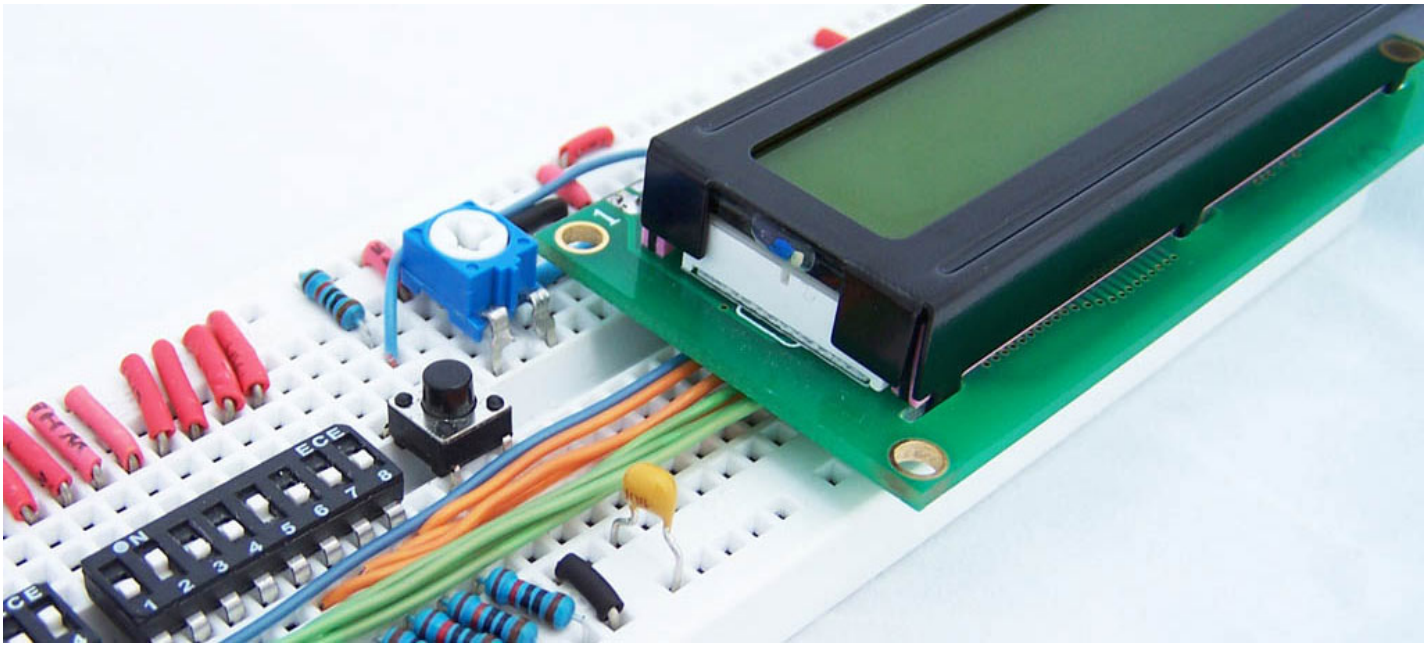
1. **Atmega8 breadboard circuit – Part 1 of 3 – Power supply**
2. **ATmega8 breadboard circuit – Part 2 of 3 – The Microcontroller**

#ATmega8 *#AVR* *#Breadboard*

PREVIOUS ARTICLE

ATmega8 breadboard circuit – Part 2 of 3 – The Microcontroller

RELATED NEWS



HD44780 CHARACTER LCD DISPLAYS – PART 1

March 9, 2010

Introduction LCD character displays can be found in espresso machines, laser printers, children's toys and...

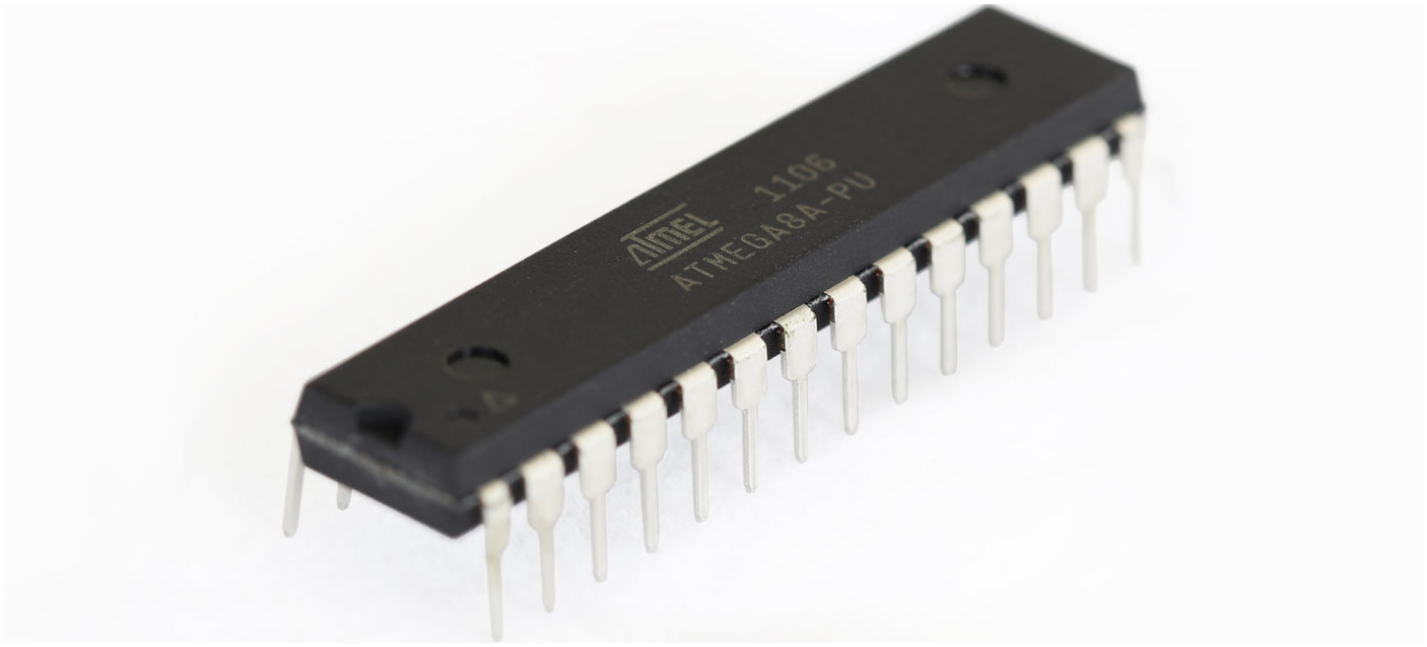
READ MORE +

SEVERED HAND IN A JAR HALLOWEEN DISPLAY

November 8, 2009

For this post I thought I'd try something a little different. I've created 3 videos...

READ MORE +



ATMEGA8 BREADBOARD CIRCUIT – PART 2 OF 3 – THE MICROCONTROLLER

July 26, 2009

This tutorial continues on from ATmega8 Breadboard Circuit - Part 1 where we build a...

25 COMMENTS

Anonymous August 15, 2009 at 12:00 am

great. thanks

Log in to Reply

Karl August 18, 2009 at 12:00 am

Thanks for these tutorials. I found them very clear and easy to follow. Hope you continue to create more! Love to see a tutorial on C programming for embedded devices (AVR based). Especially the setting up of Ports etc. Coming from Arduino land, this is a bit of a black art in my opinion??

Log in to Reply

R Bolijn February 6, 2011 at 1:04 am

Hi all,

Great tutorial. I have a problem though:

The .hex file seems to have loaded to the mcu just fine. I get the same responses from the cmd window as in this tutorial, but there's one extra line:
warning: cannot set sck period. please check for usbasp firmware update (after the \erasing chip\ line).
Does aneone know what to do? The led is flashing.

Ronald

Log in to Reply

complaining about the '&=' part.
Perhaps I need a path to the libararies?

Log in to Reply

Flywheel88 February 24, 2011 at 11:56 pm

I have found the solution to the problem I found with the code.
If you copy/paste the code from the web page, line 47 reads as:
47. PORTC & a m p ; = ~_BV(PC4);
(I added extra spaces because somehow this 'amp;' part appears and disappears)
The line should be:
47. PORTC &= ~_BV(PC4);
This works fine! The led flashes now.
Thanks for the clear instructions.

Log in to Reply

Fernando Denis Emert November 18, 2015 at 9:41 am

I love you XD

Log in to Reply

NiveusLuxLucis July 13, 2011 at 1:28 pm

I keep getting this:

avrdude: warning: cannot set sck period. please check for usbasp firmware update
.
avrdude: error: programm enable: target doesn't answer. 1
avrdude: initialization failed, rc=-1
Double check connections and try again, or use -F to override
this check.

When I'm using a straight off the shelf ATMEGA48A-PU. The user guide says something about an SCK option, but it doesn't say how to set it. Can anyone offer a solution?

Rival August 3, 2014 at 3:54 am

Same here, I'm using a cheapo programmer from eBay.

Log in to Reply

NiveusLuxLucis July 13, 2011 at 8:02 pm

ignore that I got it working

Log in to Reply

Peertux May 29, 2012 at 8:46 am

I just love it when people find a solution for their problem and say "SOLVED" or something like that, without posting the solution for others.

Log in to Reply

ibnu tohid October 22, 2011 at 4:23 pm

hi all..

i keep on getting this error.. can anyone show me how to correct this.

avrdude: warning: cannot set sck period. please check for usbasp firmware update

avrdude: error: programm enable: target doesn't answer. 1
avrdude: initialization failed, rc=-1
Double check connections and try again, or use -F to override
this check.

i already check the connection several times and i dont find any problem with it. really appreciate your help. thanks in advance

Log in to Reply

Log in to Reply

ibnu tohid October 25, 2011 at 3:22 pm

not yet. i can't even write the program to the device

Log in to Reply

Daniel Garcia Author October 27, 2011 at 5:03 pm

Are you using a breadboard, a protostack AVR board or something else?

If you want, send me a message via our contact us page http://www.protostack.com/index.php?main_page=contact_us and I'll help you through the issue.

Log in to Reply

Mike October 10, 2015 at 10:01 pm

I had the same issue and it was caused from the USBasp not providing power to the breadboard. All i have to do was to close a jumper on the programmer to supply 5V to the board.

Log in to Reply

John R. Hartmus II January 18, 2012 at 8:09 am

I see that the USBASP programmer supplies +5 VDC on pin 2, and in the picture which shows your breadboard connected to your laptop via your USBASP programmer, you are obviously using the programmer as your power source. Would I be safe to not connect pin 2 from the programmer, and use another 5 VDC power supply for the MCU, as long as the grounds are connected?

Daniel Garcia Author January 19, 2012 at 7:29 am

There is a jumper on the USBASP that controls the power output on pin 2. On the 2 latest versions of the programmer that jumper is labeled as J1 or JP1. When set, this jumper allows power to go to pin 2. Unset this jumper if you want to have a different power source to the MCU.

Log in to Reply

Thoquz July 6, 2012 at 7:10 am

I made this on my breadboard and it gave me this error:
“avrdude: initialization failed, rc=-1”

I fixed this by building a standalone target board that uses the 6 pin icsp connector rather than the 10 pin one used here.

As mentioned in the comments the code in the above tutorial has a syntax error.
Here is a link to my version of that code that works. (http://pastebin.com/jDm2maeM)

Log in to Reply

wowme@wtf.com August 31, 2012 at 9:31 pm

@Thoquz
I had the “‘avrdude: initialization failed, rc=-1”’ error on another project using a usbtiny programmer.

My fix was to change the avrdude command line from -B 1 to -B 32.
Some smaller value than 32 Might Have fixed it, but the first fix was used.

Log in to Reply

Mohan November 23, 2013 at 3:54 pm

After make programe commond screen dispys
No rule to make target
In note pad [WinAVR} Programe I got following massage

> Process Exit Code: 2
> Time Taken: 00:00
Pl reply how to execute programme
Log in to Reply

shekar July 17, 2014 at 3:16 am

I keep getting this:

avrdude: warning: cannot set sck period. please check for usbasp firmware update
.
avrdude: error: programm enable: target doesn't answer. 1
avrdude: initialization failed, rc=-1
Double check connections and try again, or use -F to override
this check.

Log in to Reply

Rival August 3, 2014 at 3:53 am

Same here...

Log in to Reply

Rival December 11, 2014 at 4:09 am

I realized later that I had the wiring wrong on my programmer and the way it was connected to the chip.
I gave up on using avrdude anyways and now use Khazama AVR Programmer or eXtreme Burner, had less issues using it. It's graphical as well and easier to use.

Log in to Reply

Daithi July 16, 2015 at 11:37 am

Ivan Fomin September 14, 2015 at 5:58 pm

Really cool basic tutorial, something that I search for a long time. And it really works! Even with my very old circuit that I bought 3 years ago.

Log in to Reply

LEAVE A REPLY

You must be logged in to post a comment.

CONTACT US +

STAY CONNECTED +

INFORMATION +

CUSTOMER CARE +

PAYMENT OPTIONS +