

Digistump Forums

[Back to Digistump.com](#)

Welcome, **Guest**. Please login or register.
Did you miss your activation email?

Forever

Login with username, password and session length

[Home](#) [Help](#) [Search](#) [Login](#) [Register](#)

Digistump Forums » The Digispark » Digispark (Original) Support » Adding the bootloader to my digispark clone

« previous next »

Pages: [1]

 **Author**

Topic: Adding the bootloader to my digispark clone (Read 9132 times)

albercook

Newbie



Posts: 33



Adding the bootloader to my digispark clone

« **on:** May 18, 2014, 03:03:18 pm »

OK I think I understand what I have to do but want to run it by all you bright people. I have been reading up in this but feel free to point out relevant posts that I may have missed. I have built a small board with a tiny85 and some additional circuitry. It works when I solder on a tiny85 that I have removed from a digispark so I didn't make any critical errors.

Now I want to build more with tiny85's from digikey that don't come with the bootloader installed. My board is very small and I did not have room for a jumper like JP3 on the digispark so I plan to program the chips before I solder them on my board. If I understand this correctly ironically I'm going to use an in system programmer to program chips out of my system.

If I understand correctly I need to connect as follows.

MOSI (Pin1) from the programmer to MOSI (PB0) on the chip
MISO (Pin9) from the programmer to MISO (PB1) on the chip
RES (Pin5) from the programmer to RESET (PB5) on the chip
SCK (Pin7) from the programmer to SCK (PB2) on the chip
VCC (VTG Pin2) from the programmer to VCC on the chip
GND (Pin4,6,8,10) from the programmer to GND on the chip

Am I on the right track? Do I have to worry about fuse bits?

Thanks,
George

« Last Edit: May 24, 2014, 11:44:40 am by albercook »

 Logged

defragster

Sr. Member



Re: Adding the bootloader to my digispark clone

« **Reply #1 on:** May 18, 2014, 04:18:30 pm »



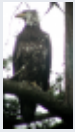
Posts: 467

Do a forum "Search results for: mosi miso" and let me know if any of them have the info to get you running. I got 11 results including a hit on this entry.

Logged

defragster

Sr. Member



Posts: 467



Re: Adding the bootloader to my digispark clone

« Reply #2 on: May 18, 2014, 05:01:31 pm »

Another posting led me to this: <http://www.instructables.com/id/Digispark-DIY-The-smallest-USB-Arduino/?ALLSTEPS> that should be what you/we need!

Logged

gogol

Sr. Member



Posts: 398



Re: Adding the bootloader to my digispark clone

« Reply #3 on: May 19, 2014, 01:40:47 am »

http://digistump.com/wiki/digispark/tutorials/programming#using_littlewire_as_us

Logged

albercook

Newbie



Posts: 33



Re: Adding the bootloader to my digispark clone

« Reply #4 on: May 21, 2014, 09:21:39 am »

Thanks defragster. I think there is one major thing that I don't understand. Can I install the bootloader through the USB connector on the digispark (that is through the zener diodes and resistors) or do I have to go directly to the pins on the chip?

Logged

gogol

Sr. Member



Posts: 398



Re: Adding the bootloader to my digispark clone

« Reply #5 on: May 22, 2014, 12:09:24 am »

That is all answered in the Wiki. Just the next chapter in that document I linked above:

http://digistump.com/wiki/digispark/tutorials/programming#creating_digispark_clone

You need one device as ISP-Programmer to flash the initial bootloader to the chip.

Once the boot-loader is on the device, the boot-loader is able to update itself via USB, but than you need to use the micronucleus binary!

Logged

albercook

Newbie



Posts: 33



Re: Adding the bootloader to my digispark clone

« Reply #6 on: May 24, 2014, 09:22:37 am »

Thanks for the links. I have read them it is just that sometimes I don't fully understand them. That is why is is so nice to be able to ask questions here.

Erik was kind enough to give give me the hex file that is installed at the

factory. I have the same USBisp that he used and am using AVRdude in command line. I get the following results.

Code: [Select]

```
avrdude.exe -p attin
y85 -c usbasp -U flash:w:DIGISPARK_FACTORY_FINAL.hex:i -U Ifuse:w:0xe1:m -U efuse:w:0xfe:m -U hfuse:w:0x5d:m

avrdude.exe: warning: cannot set sck period. please check for usbasp firmware u
date.
avrdude.exe: AVR device initialized and ready to accept instructions

Reading | ##### | 100% 0.03s

avrdude.exe: Device signature = 0x1e930b
avrdude.exe: NOTE: FLASH memory has been specified, an erase cycle will be per
rmed
                To disable this feature, specify the -D option.
avrdude.exe: erasing chip
```

I do get that warning about SCK period but don't think that is a problem.

The chip that I bought is ATTINY85-20SH it is an 8SOIC

To make contact I pressed the pins onto a little 8SOIC carrier board.

Finally, I soldered it onto a DIGIsPark board. (I had scavenged some programmed 85s from DigiSpark boards previously)

When I try to program it I get the "USB Device Not Recognized" error.

I tried a powered hub and the same system works fine to program a standard DigiSpark.

I'm thinking that I should not have to do the 5-pin, 6-pin steps if I'm doing it this way. Is that right?


Do I need to do something different with the fuses?

Any thoughts?

I really have read up on this I'm just not understanding something in what I have read or I'm missing something.

Thanks,
Albercook

« Last Edit: May 24, 2014, 12:18:10 pm by albercook »

 Logged

gogol

Sr. Member



Posts: 398



Re: Adding the bootloader to my digispark clone

« Reply #7 on: May 24, 2014, 11:25:36 am »

First an friendly request: Please use the CODE-Tag for code and code output, that makes it just better readable.

The biggest problem I see is, that you are using a wrong parameter **Ifuse** instead **Lfuse**!

There are three fuses, you need to care about. A HIGH and a LOW Fuse and the EXTENDED Fuse.

You are now burning the HFUSE and the EFUSE, but with a messed up LFUSE, so your clock is too slow!

You should follow the recipe, not to disable RESET in your first shots, as with that you render your attiny unusable, if something is not correct, as the LFUSE in your example.

With disabling reset in the HFUSE, you need now HV-Programmers to change the fuses, as ISP-programmers need reset!

So this settings from the linked tutorial:

Code: [Select]

```
attiny85-RESET-enabled.bootloader.low_fuses=0xf1
attiny85-RESET-enabled.bootloader.high_fuses=0xdf
attiny85-RESET-enabled.bootloader.extended_fuses=0xfe
```

are the best fuses for this kind of try, as you can repeat it how often you need.

A good fuse-calculator is here: <http://www.engbedded.com/fusecalc>

Good luck for the next try!

 Logged

albercook

Newbie



Posts: 33



Re: Adding the bootloader to my digispark clone

« **Reply #8 on:** May 24, 2014, 12:45:40 pm »

Thanks again. I changed the post. Now it has the code Tags. I looked at the fuse calculator but for a noob like me I have to look up all of the fuses and read about them. Even then I usually have more questions after reading and still am not sure which way to set the fuses. Not that I'm unwilling to learn.

Thanks very much. I will keep you posted.

 Logged

albercook

Newbie



Posts: 33



Re: Adding the bootloader to my digispark clone

« **Reply #9 on:** May 24, 2014, 02:14:03 pm »

I think is actually worked.

Code: [Select]

```
e:w:0xdd:m

avrdude.exe: warning: cannot set sck period. please check for usbasp firmware u
date.
avrdude.exe: AVR device initialized and ready to accept instructions

Reading | ##### | 100% 0.03s

avrdude.exe: Device signature = 0x1e930b
avrdude.exe: NOTE: FLASH memory has been specified, an erase cycle will be per
rmed

        To disable this feature, specify the -D option.
avrdude.exe: erasing chip
avrdude.exe: warning: cannot set sck period. please check for usbasp firmware u
date.
```

I played a some with the fuse calculator. I think I understand a bit more.

Ha Ha bit

It worked. I was able to program is like a regular DigiSpark.

Next I will change the one fuse so that I can use the reset pin.

Thanks

« Last Edit: May 26, 2014, 03:08:01 pm by albercook »

 Logged

albercook

Newbie



Posts: 33



Re: Adding the bootloader to my digispark clone

« Reply #10 on: May 26, 2014, 03:11:37 pm »

Thanks for all the help. Here is the AVRdude command line I ended up using in the end. It is the one that Erik sent me.

Code: [Select]

```
avrdude.exe -p attiny85 -c usbasp -U flash:w:FINAL.hex:i -U lfuse:w:0xe1:m -U efu:
```

I changed blink to blink pin 5 and it worked so the reset must be disabled as expected.

Thanks again for all the help.

George

 Logged

gogol

Sr. Member



Posts: 398



Re: Adding the bootloader to my digispark clone

« Reply #11 on: May 27, 2014, 12:01:49 am »

[Corrected on 2014-05-30]

For all others I recommend first using **0xDF** for the HIGH FUSE, which keeps reset enabled.

That can be repeated as often as needed, the resulting digispark clone works like an digispark with the only difference, that PIN5 (RESET) is not available as IO.

When everything works fine, you can change the HIGH FUSE to **0x5F**, which disables reset, allowing to use that PIN as an additional IO-pin.

From that moment however, ISP-programming is not longer possible. When the bootloader is not working properly, you need an High Voltage Serial Programmer to fix the settings.

So the recommended way is:

Write the bootloader to the chip and test (repeat as often as needed):

Code: [Select]

```
avrdude.exe -p attiny85 -c usbasp -U flash:w:FINAL.hex:i -U lfuse:w:0xF1:m -U hfi
```

Finally change the HIGH FUSE, so reset is disabled. (and the ISP-programmer is locked out)

Code: [Select]

```
avrdude.exe -p attiny85 -c usbasp -U hfuse:w:0x5F:m
```

The above Flags are those, which are documented in the `./hardware/digispark/boards.txt` file of the digispark IDE and is used in many threads of this forum.

Those you are using should work also, the differences are:

LOW: 0xF1 / 0xE1 The original setting is configured for slow rising power, your setting is used for a fast rising power (compare Table 6-5 of the attiny85 documentation)

14CK + 16K (16384) CK + 4 ms instead of **14CK + 1K (1024) CK + 4 ms**

HIGH: 0x5F / 0x5D The original setting has no Brown Out Detection (BOD), yours is triggered below 2.7V

« Last Edit: May 30, 2014, 01:21:48 am by gogol »

 Logged

albercook

Newbie



Posts: 33



Re: Adding the bootloader to my digispark clone

« **Reply #12 on:** May 28, 2014, 12:47:33 pm »

Great suggestion. I will do it that way. In addition to allowing code changes it may also help with poor connections. I'm programming the surface mount versions. I press them down to make contact. I have been worried that I might have a bad connection and end up with a bad downloader and no reset capability. This way I can confirm a good download before I convert the reset pin to an I/O pin.

Thanks.

 Logged

albercook

Newbie



Posts: 33



Re: Adding the bootloader to my digispark clone

« **Reply #13 on:** May 28, 2014, 01:23:08 pm »

Please correct me if I am wrong but I think the two fuse options that we are comparing are only change in the hfuse. I think (if I understand the fuse calculator, which I most certainly don't) in both cases the lfuse=0xE1 and efuse=0xFE.

hfuse=0xDD leaves pin5 as reset.

hfuse=0x5D makes pin5 an I/O and further reprogramming will require a high voltage programmer.

I hope this is correct. I wrote all the fuses down and double checked it.

Thanks again for all the help gogol and others.

 Logged

gogol

Sr. Member



Posts: 398

**Re: Adding the bootloader to my digispark clone**« **Reply #14 on:** May 30, 2014, 01:23:58 am »

Sorry, looks like, that I messed something up in my answer.
I should have taken more time, to read, what I have written. But I see, that you are now really on the right track!
Good Luck!



Logged

Pages: [1]

PRINT

« previous next »

[Digistump Forums](#) » [The Digispark](#) » [Digispark \(Original\) Support](#) » Adding the bootloader to my digispark clone

Jump to: => Digispark (Original) Support ▼ go

Back to Digistump.com

© 2013 Digistump LLC

Powered by SMF 2.0.11 | SMF © 2015, Simple Machines

[XHTML](#) [RSS](#) [WAP2](#)