

Загрузка ATmega88 для Arduino [с кодом]

👤 : Ройс < <https://www.morethantechnical.com/author/roys/> >

4 апреля 2012 г. <

📅 <https://www.morethantechnical.com/2012/04/04/bootloading-an-atmega88-for-arduino-w-code/> >

Просто хотел выложить небольшую заметку о том, как установить загрузчик Arduino на ATmega88.

Я очень приблизительно следовал:

<http://www.ladyada.net/learn/avr/avrdude.html> <

<http://www.ladyada.net/learn/avr/avrdude.html> >

И

<http://itp.nyu.edu/physcomp/Tutorials/ArduinoBreadboard> <

<http://itp.nyu.edu/physcomp/Tutorials/ArduinoBreadboard> > (Хотя они делают это слишком сложным)

И техническое описание ATMEL для ATmega88 также очень полезно иметь всегда открытым:

<http://www.atmel.com/Images/2545s.pdf> Во- <

<http://www.atmel.com/Images/2545s.pdf> >

первых, положите все на макет: (Это заняло у меня некоторое время... Спасибо Д.Меллису и М.Фельдмайеру !!)



([Фритцинг](http://fritzing.org/) < <http://fritzing.org/>> потрясающий)

Примечание:

- при программировании с помощью AVR ISP mkII отключите светодиод, поскольку он замыкается на землю.
- Убедитесь, что вы помните о подтягивающем резисторе при сбросе, чтобы сигнал был хорошим и высоким (спасибо Б.Мелтону!).,
- а также конденсатор емкостью 0,1 мкФ для USB-программатора FTDI, так что короткий всплеск сброса становится хорошим длинным сигналом, так что чип будет доволен.

Теперь нам нужно разобраться в предохранителях на чипе, чтобы он:

- Запустите загрузчик после СБРОСА (BOOTRST fuse)
- Установите размер раздела загрузчика равным 1024 словам (2048 байт), поскольку скомпилированный загрузчик составляет ~ 1400 байт (BOOTSZ0 и BOOTSZ1).
- Включите программирование SPI, конечно, мы не хотим кирпичить чип (SPIEN).
- Установите часы на внутреннюю частоту 8 МГц, так как мы не используем внешнюю частоту

Чтобы получить значения для предохранителей:

<http://www.engbedded.com/fusecalc/> <
<http://www.engbedded.com/fusecalc/>>

В итоге я использовал:

```
EFUSE = 0x00  
LFUSE = 0xe2  
HFUSE = 0xdf
```

Следующий шаг - написать код загрузчика. Я позаимствовал ATmegaBOOT_168.c от Arduino 1.0, в котором уже есть все настройки, но он не поддерживает ATmega88 ... потребовались некоторые изменения.

Но вы, ребята, можете получить конечный продукт:

https://raw.githubusercontent.com/royshil/ATmega88-bootloader/master/ATmegaBOOT_88.c <

https://raw.githubusercontent.com/royshil/ATmega88-bootloader/master/ATmegaBOOT_88.c>

Вы также можете просто получить шестнадцатеричный файл и сразу же записать его!

https://raw.githubusercontent.com/royshil/ATmega88-bootloader/master/ATmegaBOOT_88_m88.hex <

https://raw.githubusercontent.com/royshil/ATmega88-bootloader/master/ATmegaBOOT_88_m88.hex>

Чтобы собрать его, я также позаимствовал Makefile Arduino, но он также нуждался в некоторой доработке:

https://raw.githubusercontent.com/royshil/ATmega88-bootloader/master/arduino_Makefile <

https://raw.githubusercontent.com/royshil/ATmega88-bootloader/master/arduino_Makefile>

(Обратите особое внимание на “LDSECTION = -section-start =.text = 0x1800”, потому что он содержит ключ к установке загрузчика в нужное место на флэш-памяти. Я работал в соответствии с таблицей данных, посмотрите таблицу 26-9, стр. 281)

, Но затем наступил момент, когда что-то начало работать ... AVR ISP mkII больше не жаловался, что не может общаться с чипом и сжег загрузчик!!

Сделать Arduino счастливым

Загрузчик ОК сгорел на чипе, теперь мне нужно было настроить Arduino для хорошей работы с ATmega88.

Я изменил \$ (ARDUINO) /hardware/arduino/boards.txt чтобы добавить это:

```
3 atmega88.upload.maximum_size=7168
4 atmega88.upload.speed=19200
5 atmega88.bootloader.low_fuses=0xe2
6 atmega88.bootloader.high_fuses=0xdf
7 atmega88.bootloader.extended_fuses=0x00
8 atmega88.bootloader.path=atmega88
9 atmega88.bootloader.file=ATmegaBOOT_88_m88.hex
10 atmega88.bootloader.unlock_bits=0x3F
11 atmega88.bootloader.lock_bits=0x0F
12 atmega88.build.mcu=atmega88
13 atmega88.build.f_cpu=8000000L
14 atmega88.build.core=arduino
15 atmega88.build.variant=standard
```

Теперь Arduino видит его в разделе Tools -> Board -> ATmega88
Кроме того, не забудьте поместить шестнадцатеричный файл загрузчика в: \$ (ARDUINO)/hardware/arduino/
bootloaders/atmega88/ATmegaBOOT_88_m88.hex

На этом этапе вы должны иметь возможность загружать эскизы Arduino на свой чип ATmega88.

Первый скетч, который был успешно загружен, был для меня как чудесная песня о русалке 😊

Получить весь код

<https://github.com/royshil/ATmega88-bootloader> <

<https://github.com/royshil/ATmega88-bootloader>>

Для программирования через USB я использую:

<http://arduino.cc/en/Main/USBSerial> <

<http://arduino.cc/en/Main/USBSerial>>

Я использовал это руководство примерно для настройки приложения Arduino для “понимания” ATmega88:

<http://arduino.cc/en/Hacking/Programmer> <

<http://arduino.cc/en/Hacking/Programmer>>

Это также полезно:

<http://arduino.cc/en/Hacking/Bootloader> <

<http://arduino.cc/en/Hacking/Bootloader>>

(Теперь у нас в МТТ появилась новая категория – “электроника”!

Я надеюсь скоро поделиться в нем кое-чем интересным ...)

Русский

Перевести картинки

© 2022 Более чем технический <
<https://www.morethantechnical.com/>>

Вверх ↑