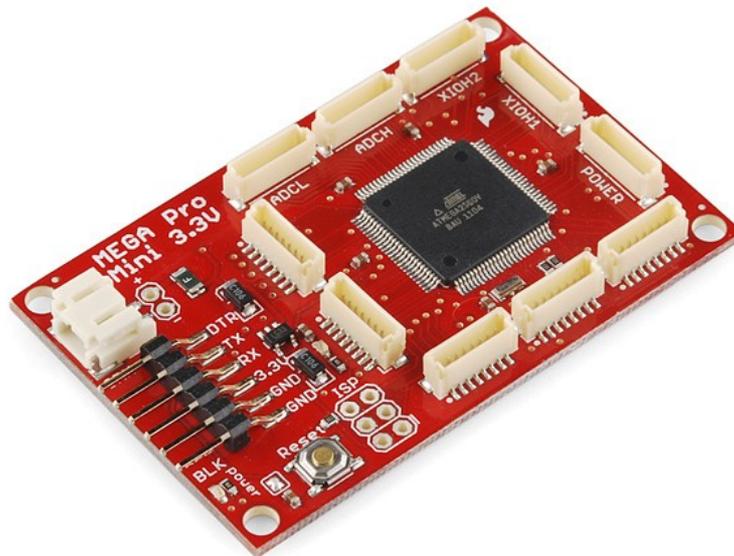


# SPARKFUN

## Mega Pro Mini - 3.3V



### Введение

Если вам необходим функционал Arduino Mega в миниатюрном исполнении, то эта плата - то, что вы искали. Серия “Pro” Arduino – совместимых контроллеров пополнилась новой моделью: “Mega Pro Mini”. Эта версия платы питается напряжением 3,3В, что возможно благодаря версии “бутлоадера” stk500v2, реализующей работу микроконтроллера ATmega 2560 на частоте 8 МГц. Для того, что бы минимизировать размеры платы, была изменена разводка платы, а стандартные разъемы были заменены более компактными JST.

Серия “Pro” предназначенная для продвинутых пользователей, предпочитающих гибкое и многофункциональное устройство для своих проектов. Поэтому на плате не предусмотрен разъем USB и по тем же причинам отсутствует гнездо для подключения блока питания. Устройство подключается к компьютеру через модуль с микросхемой FT232, под которую предусмотрен шестиконтактный разъем.

**Замечание:** для программирования данного устройства в среде Arduino IDE, были внесены изменения в “бутлоадер”, чтобы он мог поддерживать системы, работающие от напряжения 3,3В. Вам необходимо скачать и установить файлы определения платы, после чего с ней можно будет работать как с Arduino Mega. Данный архив содержит все необходимые вам файлы, включая “readme”, в котором содержится информация по установке:

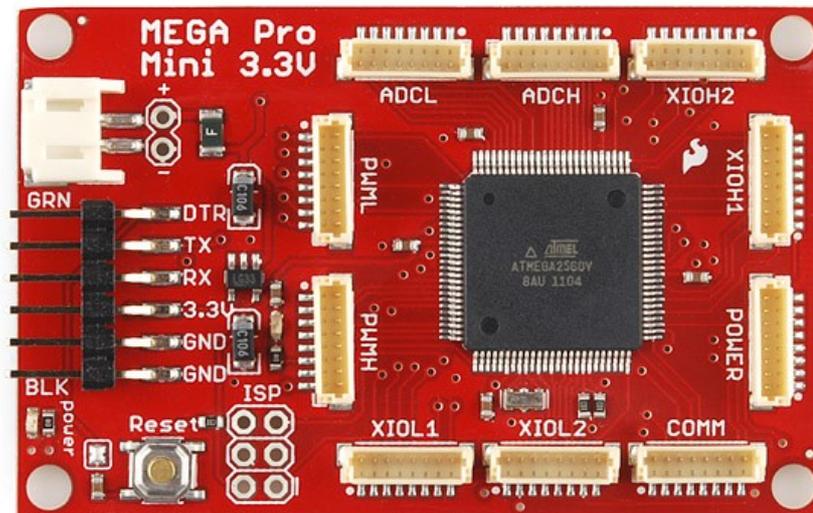
[http://dlnmh9ip6v2uc.cloudfront.net/datasheets/Dev/Arduino/Boards/mega\\_pro\\_3.3V.zip](http://dlnmh9ip6v2uc.cloudfront.net/datasheets/Dev/Arduino/Boards/mega_pro_3.3V.zip)

(Arduino IDE 0.22)

[http://dlnmh9ip6v2uc.cloudfront.net/datasheets/Dev/Arduino/Boards/mega-pro-3.3V-v10\\_.zip](http://dlnmh9ip6v2uc.cloudfront.net/datasheets/Dev/Arduino/Boards/mega-pro-3.3V-v10_.zip)

(Arduino IDE 1.0)

Так же вы можете использовать среду [Wiring](#), в которой данная плата официально поддерживается.



В настоящее время Arduino IDE доступно для Windows, Mac OS X и Linux 32bit. Среду можно скачать по ссылке: <http://arduino.cc/en/Main/Software>

Данная программа распространяется в виде архива и не нуждается в установке.

Перед началом программирования необходимо выбрать номер последовательного порта (виртуальный COM – порт) и тип платы. Сделать это можно во вкладке **Service**.

В меню **File** -> **Examples** находятся базовые примеры работы с портами Arduino и его библиотеками.

## Структура программы

Любая программа, написанная в Arduino IDE, должна содержать в себе две базовые функции: **setup()** и **loop()**. Как только вы подаете питание на плату - запускается функция **setup()**. Этот факт делает ее очень удобной для хранения в ней нужных пользователю настроек микроконтроллера - конфигурации прерываний, портов и т.д.

Функция **loop()** является основой программы. Она исполняется сразу после функции **setup()** и ее содержимое будет постоянно повторяться, пока устройство не будет выключено или перезагружено.

Информация о языке программирования можно найти здесь:

<http://arduino.cc/en/Reference/HomePage> (на английском языке)

<http://arduino.ru/Reference>

## Примеры

Начнем с самого простого примера, с которого обычно начинают обучение программированию плат Arduino.

```
/*
```

```
Пример1. Blink
```

```
Включает светодиод на секунду, затем выключает его на секунду.
```

```
*/
```

```
// На большинстве плат Arduino к пину 13 подсоединен светодиод.
```

```
//Для удобства, дадим ему название:
```

```
int led = 13;
```

```
//функция setup() запускается один раз:
```

```
void setup()
```

```
{
```

```
//инициализация цифрового порта 13 как выхода.
```

```
pinMode(led, OUTPUT);
```

```
}

//тело функции loop() повторяется в бесконечном цикле:
void loop()
{
  digitalWrite(led, HIGH); // включить светодиод (HIGH - уровень напряжения)
  delay(1000);             // подождать секунду
  digitalWrite(led, LOW); // выключить светодиод
  delay(1000);            // подождать секунду
}

/*
Пример2. ReadAnalogVoltage
Считывает значение на ножке D0, преобразует его в напряжение и выводит результат в
монитор последовательного порта. Присоедините среднюю ножку потенциометра к
порту A0, а боковые ножки к +5 и общему проводу.
*/

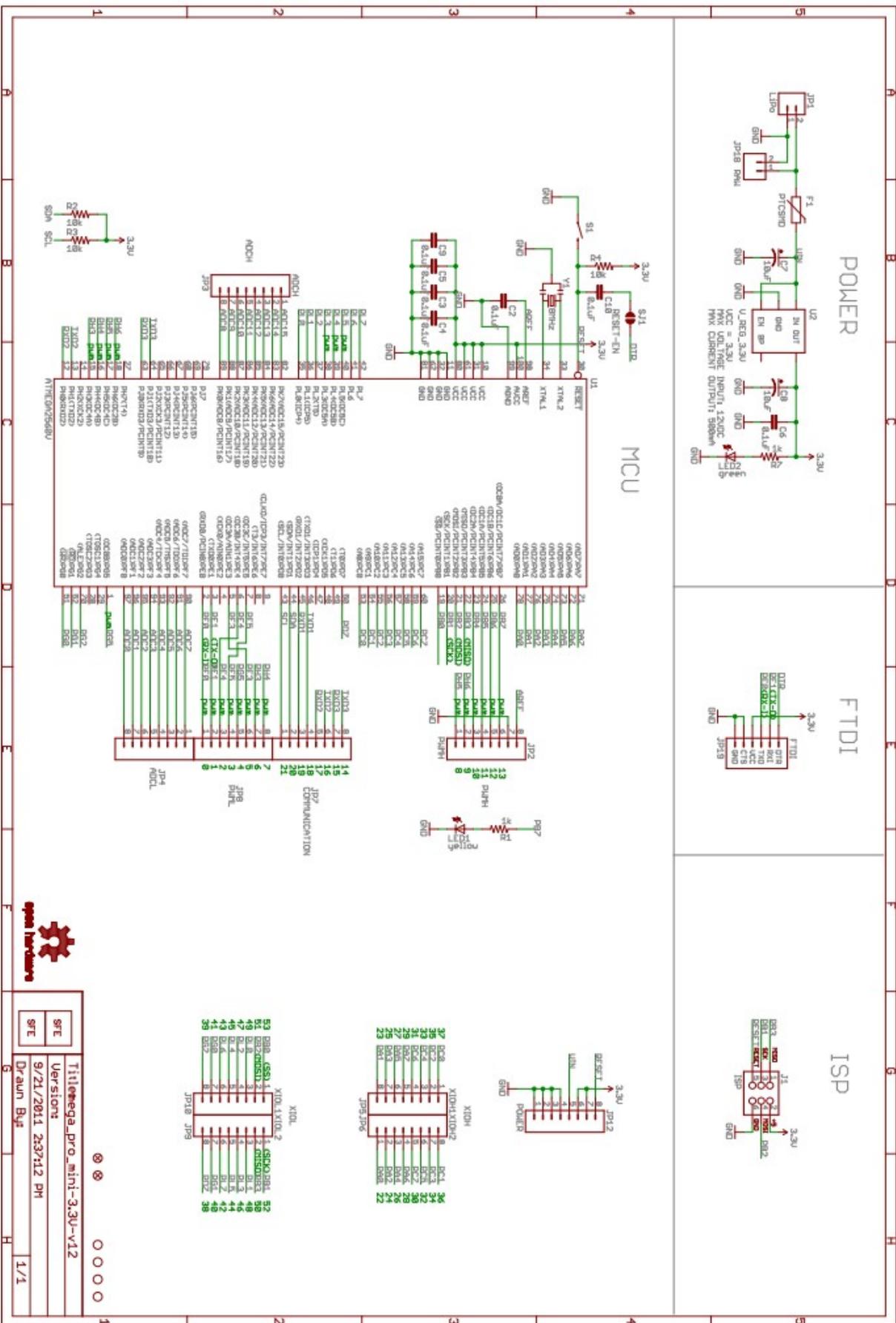
void setup()
{
  // инициализация последовательного порта микроконтроллера на скорости 9600 //бит в
секунду:
  Serial1.begin(9600);
}

//тело функции loop() повторяется в бесконечном цикле:
void loop()
{
  // считать значение аналогового входа 0:
  int sensorValue = analogRead(A0);
  // Преобразовать значение АЦП (которое лежит в диапазоне 0 – 1023) в напряжение (0 //-
5V):
  float voltage = sensorValue * (5.0 / 1023.0);
  // вывести считанное значение:
  Serial1.println(voltage);
}
```

## **Характеристики платы**

- Микроконтроллер ATmega2560
- Разъем USB удален с платы
- Рабочее напряжение 3,3В
- Нет необходимости в специальных преобразователях для питания внешних устройств (GPS, акселерометры, сенсоры и т.д.)
- Входное напряжение (рекомендуется) 7-12В
- Входное напряжение (Предел) 6-20В
- Наличие на плате линейного стабилизатора напряжения на 3,3В
- Защита от перегрузки
- Защита от подключения питания неверной полярности
- Самовосстанавливающийся предохранитель защищает от короткого замыкания
- Ток на один вход/выход 40 мА
- Тактовая частота 8 МГц (кварцевый резонатор)

## **Принципиальная схема**



Mega Pro Mini Pinout Guide		Pin Number (Pin 1 is indicated on the board by a white dot)							
		1	2	3	4	5	6	7	8
		Wire Color if you're using the Mega Pro Mini Wires from the SparkFun site							
		White	Grey	Purple	Blue	Green	Yellow	Orange	Brown
Connector Label	ATMega Pin [Arduino Pin Configuration]								
ADCL	ADC7 [A7]	ADC6 [A6]	ADC5 [A5]	ADC4 [A4]	ADC3 [A3]	ADC2 [A2]	ADC1 [A1]	ADC0 [A0]	
ADCH	ADC15 [A15]	ADC14 [A14]	ADC13 [A13]	ADC12 [A12]	ADC11 [A11]	ADC10 [A10]	ADC9 [A9]	ADC8 [A8]	
XIOH2	PA0 [22]	PA2 [24]	PA4 [26]	PA6 [28]	PC7 [30]	PC5 [32]	PC3 [34]	PC1 [36]	
XIOH1	PC0 [37]	PC2 [35]	PC4 [33]	PC6 [31]	PA7 [29]	PA5 [27]	PA3 [25]	PA1 [23]	
POWER	GND	GND	GND	GND	VIN	3.3V	3.3V	RESET	
COMM	SCL [21]	SDA [20]	RXD1 [19]	TXD1 [18]	RXD2 [17]	TXD2 [16]	RXD3 [15]	TXD3 [14]	
XIOL2	PB1 [52]	PB3 [50]	PL1 [48]	PL3 [46]	PL5 [44]	PL7 [42]	PG1 [40]	PD7 [38]	
XIOL1	PB0 [53]	PB2 [51]	PL0 [49]	PL2 [47]	PL4 [45]	PL6 [43]	PG0 [41]	PG2 [39]	
PWMH	PH5 [8]	PH6 [9]	PB4 [10]	PB5 [11]	PB6 [12]	PB7 [13]	N/C	AREF	
PWML	PE0 [0]	PE1 [1]	PE4 [2]	PE5 [3]	PG5 [4]	PE3 [5]	PH3 [6]	PH4 [7]	