Atmega32u4 Breakout

Created by lady ada



Last updated on 2016-09-12 05:54:42 PM UTC

#### **Guide Contents**

Guide Contents	2
Intro	3
About the Atmega32u4 Breakout board+	3
Why not use a Teensy	4
Assembly	6
Design	9
Design Specifications	9
Microcontroller	9
Power	9
Pinout	9
USB Development	10
Using with AVRDude	12
AVR109 Bootloader & AVRdude	12
Arduino IDE Setup	14
https://adafruit.github.io/arduino-board-index/package_adafruit_index.json	15
Using with Arduino	17
Using it with Teensyduino	19
Download	22
Download	22
Schematic	22
Fabrication Print	23

## Intro



#### About the Atmega32u4 Breakout board+

We like the AVR 8-bit family and were excited to see Atmel upgrade the series with a USB core. Having USB built in allows the chip to act like any USB device. For example, we can program the chip to 'pretend' it's a USB joystick, or a keyboard, or a flash drive! Another nice bonus of having USB built in is that instead of having an FTDI chip or cable (like an Arduino), we can emulate the serial port directly in the chip. This costs some Flash space and RAM space but that's the trade-off.

The only bad news about this chip is that it is surface mount only (SMT), which means that it is not easy to solder the way the larger DIP chips are. For that reason, we made a breakout board. The board comes with some extras like a fuse, a 16mhz crystal, USB connector and a button to start the bootloader.

#### Why not use a Teensy

We also carry a similar board from PJRC called the <u>Teensy</u> (http://adafru.it/alH). The Teensy uses the same chip so you may be wondering, why did we design a different-but-still-basically-the-same board? We've used the Teensy in a few projects and like it a lot but there are a few details that we wanted to change. We wanted...

- ...a bootloader that would work with avrdude since that is our preferred software
- ...another LED, for power-good indication
- ...a 500mA fuse on the USB power-source pin
- ...mounting holes so we could attach it easily
- ...reprogramming ISP header so we could program the board directly without the bootloader
- ...a larger reset button
- ...all the pins broken out for use with a breadboard
- ...open source bootloader that works with AVRdude

That doesn't in any way mean that it is better or replaces the Teensy. Here are some reasons we will still use the teensy for many projects

- It's really really small. A third the size of this breakout
- The bootloader that is programmed in uses only 512 bytes (instead of 2K)
- It works nicely with Teensyduino and auto-resets right before programming



## Assembly



The breakout is almost completely assembled. We don't solder on the header or 6-pin ISP connector since we find it's easy to do but hard to undo, in case you want to put different headers on, or just connect up wires directly.

The board ships with a strip of header, you can simply break the header into two pieces and place them in a breadboard.





# Design

If you're using windows, you will need an **inf** driver file, you can download it below (in the **Downloads** section) (http://adafru.it/lf8)

# **Design Specifications**

This breakout board is designed to make it easy for you to get started with the Atmega32u4, a USB-native 8 bit AVR microcontroller. We tried to keep the design simple, while taking care of all the details so that you can focus on your project's firmware and hardware.

#### Microcontroller

The main chip is an ATmega32u4, an 8 bit AVR-core processor which has the bonus of a USB core built in. This makes writing USB native programs such as serial ports, mouse/keyboard, mass-storage-controller, MIDI, etc very easy. The chip has 32K of flash and 2.5K of RAM. The microcontroller is clocked at 16 MHz with an on-board crystal.

#### Power

The board is powered by the USB port at 5V. There is a 500mA polyfuse to protect your computer from a shorted circuit. If you'd like to run the board off of another voltage, you can do that by cutting the **VCC** solder jumper bridge underneath and connecting an external voltage to pin #2 of the ISP header. Note that running the board at 3.3v @16mhz is considered overclocking. We do it for prototypes and it seems to work fine but it's out of spec!

There is a 3V pin from the microcontroller's internal regulator (USB signals are at 3V) you can use this for maybe 10mA or so, it might work as a reference voltage).

#### Pinout

Pins are labeled on the silkscreen, but for additional information, Johngineer has created<u>a</u> <u>nice pinout diagram</u> (http://adafru.it/lf9) that is helpful for using this board.

### Adafruit ATMega32u4 Breakout Board

Pins mapped to Arduino Leonardo

PE6 (INT.6/AIN0 (1) D7 SS/PCINT0 D17 PCINT1/SCLK D15 PDI/PCINT2/MOSI D16 PDO/PCINT3/MISO D14 OC0B/SCL/INT0 ~D3 SDA/INT1 D2 RXD1/INT2 D0 TXD1/INT3 D1 ICP1/ADC8 A6/D4 XCK1/ICTS



# **USB Development**

The very nice thing about this chip is the USB core built in which makes USB-device development easy. What makes it even easier & better is <u>the full USB stack already written</u> for you by <u>Dean Camera</u> (http://adafru.it/lf5) . Called <u>"LUFA"</u> (http://adafru.it/lf5) , the package comes with tons of working examples for all sorts of USB devices and its completely open source. We use the firmware as a 'starting point' which we then expand upon. Please check it out and if you find it useful consider donating time (fixing bugs,

improving documentation) or funds to Dean.

## Using with AVRDude

### **AVR109 Bootloader & AVRdude**

You can tell the bootloader is active when the**red 'Boot' LED pulses/breathes** The board will then show up as a Serial or COM port, and you can use **avrdude** to program it. The 'programmer name' is **avr109** so for example, to test you should run**avrdude -p m32u4 -P COM3 -c avr109** which will initialize the bootloader.

C:\WINDOWS\system32\cmd.exe	- 🗆 🗙
	-
C:∖Documents and Settings\ladyada>avrdude -p m32u4 -P COM18 -c avr109	
Connecting to programmer: . Found programmer: Id = "LUFACDC"; type = S Software Version = 1.0; No Hardware Version given. Programmer supports auto addr increment. Programmer supports buffered memory access with buffersize=128 bytes.	
Programmer supports the following devices: Device code: 0x44	
avrdude: AVR device initialized and ready to accept instructions	
Reading   ###################################	
avrdude: Device signature = Øx1e9587	
avrdude: safemode: Fuses OK	
avrdude done. Thank you.	
C:\Documents and Settings\ladyada>	-

You can find the serial port COM# in the Device driver, or for mac/linux users rurls /dev/cu.usb\* to list all usb serial devices



The bootloader will time out eventually (after about 10 seconds) Because we are not using a USB/serial converter, bootloading is tremendously fast, we can program a full chip in under 2 seconds!

The bootloader takes up the last 4K of FLASH, so be aware that you will only have 28K instead of 32K. We have found that this isn't very constricting as 28K is still plenty. If you'd like more space, you can always use the 6-pin ISP connector and an AVR programmer (which will delete the bootloader)

If you ever want to disable the bootloader you can cut the bottom 'HWB' jumper trace. This will disconnect the 'hardware bootloader' pin, you can then use the button as a plain reset button. For the first few runs of this board we set the fuses to still use the bootloader even with the HWB jumper cut, if you want to get rid of the bootloader, please set the fuses to remove the BOOTRST fuse. Sorry!

## **Arduino IDE Setup**

The first thing you will need to do is to download the latest release of the Arduino IDE. You will need to be using **version 1.6.4** or higher for this guide.

#### Arduino IDE v1.6.4+ Download http://adafru.it/f1P

After you have downloaded and installed **v1.6.4**, you will need to start the IDE and navigate to the **Preferences** menu. You can access it from the **File** menu in *Windows* or *Linux*, or the **Arduino** menu on *OS X*.

Ś	Arduino	File	Edit	Sketc
	About A	rduino	)	
00	Preferen	ces	Ж,	
sket	Services	5	►	
<pre>void se    // pl }</pre>	Hide Arc Hide Oth Show Al	duino ners I	H೫ H೫ ℃	un onc
void la // pi	Quit Ard	uino	ЖQ	n repe

A dialog will pop up just like the one shown below.

• • •	Prefer	ences	
Sketchbook location	:		
/Users/todd/Docur	nents/Arduino		Browse
Editor language: S	ystem Default	\$	(requires restart of Arduino)
Editor font size: 10	(requires restart of Arduin	0)	
Show verbose output	t during: compilation up	load	
Compiler warnings:	None ‡		
Display line num	nbers		
✓ Verify code after	upload		
🗌 Use external edit	tor		
Check for update	es on startup		
🗹 Update sketch fi	les to new extension on save (.pd	le -> .ino)	
Save when verify	ing or uploading		
Proxy Settings			
Server (HTTP):	Port (HTTP):	8080	
Server: (HTTPS)	Port (HTTPS):	8443	
Username:	Password:		
Additional Boards Ma	anager LIRLs:		
Mana and			
/Users/todd/Library	Arduino15/preferences.txt		
(edit only when Ardu	ino is not running)		
			OK Cancel

We will be adding a URL to the new**Additional Boards Manager URLs** option. The list of URLs is comma separated, and *you will only have to add each URL once*.New Adafruit boards and updates to existing boards will automatically be picked up by the Board Manager each time it is opened. The URLs point to index files that the Board Manager uses to build the list of available & installed boards.

To find the most up to date list of URLs you can add, you can visit the list of third party board URLs on the Arduino IDE wiki (http://adafru.it/f7U). We will only need to add one URL to the IDE in this example, but **you can add multiple URLS by separating them with commas**. Copy and paste the link below into the Additional Boards Manager URLs option in the Arduino IDE preferences.

#### https://adafruit.github.io/arduino-board-index/package\_adafruit\_index.json

Preferences			×	
Settings Network				
Sketchbook location:				
C: \Users\Jadyada \Dropbox \A	rduinoSketches		Browse	
Editor language:	System Default	(requires restart of Arduino)		
Editor font size:	12			
Interface scale:	V Automatic 100 - % (requires restart of Arduing	)		
Show verbose output during:	Compilation V upload			
Compiler warnings:	None 👻			
Display line numbers				
Enable Code Folding				
Verify code after upload				
Use external editor				
Check for updates on star	rtup			
Update sketch files to new	v extension on save (.pde -> .ino)			
Save when verifying or up	ploading			
Additional Boards Manager URLs: https://adafruit.com/package_adafruit_index.json				
More preferences can be edite	ed directly in the file	Enter a comma separated list of urls		
C:\Users\adyada\AppData\Lo	cal\Arduino15\preferences.txt	· · · · · ·		
(edit only when Arduino is not	running)			

Here's a short description of each of the Adafruit supplied packages that will be available in the Board Manager when you add the URL:

- Adafruit AVR Boards Includes support for Flora, Gemma, Feather 32u4, Trinket, & Trinket Pro.
- Adafruit SAMD Boards Includes support for Feather M0
- Arduino Leonardo & Micro MIDI-USB This adds MIDI over USB support for the Flora, Feather 32u4, Micro and Leonardo using the <u>arcore project</u> (http://adafru.it/eSI).

If you have multiple boards you want to support, say ESP8266 and Adafruit, have both URLs in the text box separated by a comma (,)

Once done click **OK** to save the new preference settings. Next we will look at installing boards with the Board Manager.

## **Using with Arduino**

Once you have the Arduino IDE Set up and you've installed the board manager package, you can select the **Adafruit 32u4 Breakout** from the boards list

ile Edit Sketch To	ols Help		
Dlink §	Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload		•
modified 8 by Scott Fi */	Serial Monitor Ctrl+Shift Serial Plotter Ctrl+Shift	:+M :+L	Arduino ARM (32-bits) Boards Arduino Due (Programming Port) Arduino Due (Native USB Port)
	Board: "Adafruit 32u4 Breakout"		Adafruit Boards
<pre>// the setup void setup()    // initiali    pinMode(7,</pre>	Port: "COM21 (Adafruit 32u4 Breakout)" Programmer: "USBtinyISP" Burn Bootloader		Adafruit Flora Adafruit Bluefruit Micro Adafruit Gemma 8MHz
}		0.27	Adafruit Feather 32u4
<pre>// the loop function runs over and over again forever void loop() {</pre>		•	Adafruit 32u4 Breakout Adafruit Trinket 8MHz
digitalWrite(	7, HIGH); // turn the LED on (HIGH is the	he vo	Adafruit Trinket 16MHz

#### Then select the port

💿 Blink   Arduino 1.6.	7		
File Edit Sketch Too	ls Help		
Blink §	Auto Format Archive Sketch	Ctrl+T	
modified 8 by Scott Fi */	Serial Monitor Serial Plotter	Ctrl+Shift+M Ctrl+Shift+L	
	Board: "Adafruit 32u4 Breakout"	Þ	
// the setup	Port	•	Serial ports
<pre>void setup()     // initiali     pinMode(7,</pre>	Programmer: "USBtinyISP" Burn Bootloader	ו	COM1 COM21 (Adafruit 32u4 Breakout)

#### Create a new sketch:

 ${\ensuremath{\textit{//}}}$  the setup function runs once when you press reset or power the board

```
void setup() {
    // initialize digital pin 13 as an output.
    pinMode(7, OUTPUT);
}
// the loop function runs over and over again forever
void loop() {
    digitalWrite(7, HIGH); // turn the LED on (HIGH is the voltage level)
    delay(1000); // wait for a second
    digitalWrite(7, LOW); // turn the LED off by making the voltage LOW
    delay(1000); // wait for a second
}
```

And upload it! You should see the red LED blink once a second

Unlike modern Arduino boards, this board does not auto-reset! Before uploading, you'll need to press the RESET button to get the BOOT LED pulsing

That's it, you can now use the board from the Arduino IDE, you can use this pinmapping guide to help line up which pin is which:

### Adafruit ATMega32u4 Breakout Board

Pins mapped to Arduino Leonardo

PE6 (INT.6/AIN0 (1) D7 SS/PCINT0 D17 PCINT1/SCLK D15 PDI/PCINT2/MOSI D16 PDO/PCINT3/MISO D14 OC0B/SCL/INT0 ~D3 SDA/INT1 D2 RXD1/INT2 D0 TXD1/INT3 D1 ICP1/ADC8 A6/D4 XCK1/ICTS T1/IOC4D/ADC9 A11/D12 T0/OC4D/ADC10 A7/~D6



# Using it with Teensyduino

If you'd like to use Teensyduino type programs, you can do that too, by using the TeeOnArdu board plugin. Use the board manager, and select the TeeOnArduino package and install it

💀 Boards Manager	×
Type All 👻 tee	
Adafruit TeeOnArdu by Adafruit Boards included in this package: TeeOnArdu (Leo on TeensyCore), Flora (TeensyCore). Online help More info	Install
	Ţ
	Close

#### Then in the board menu, selectTeeOnArdu (Leonardo on TeensyCore)

TrellisMidi168   Arduino 1.6.7				Generic ESP8266 Module Adafruit HUZZAH ESP8266	
File Edit Sketch To TrellisMidi168 for (uin	ols Help Auto Format Archive Sketch Fix Encoding & Reload Serial Monitor	Ctrl+T Ctrl+Shift+M		NodeMCU 0.9 (ESP-12 Module) NodeMCU 1.0 (ESP-12E Module) Olimex MOD-WIFI-ESP8266(-DEV) SparkFun ESP8266 Thing SweetPea ESP-210	
// Ge	Serial Plotter	Ctrl+Shift+L		TeeOnArdu	
if (tr	Board: "TeeOnArdu (Leo on Teen	syCore)"	•	TeeOnArdu (Leo on TeensyCore) Flora (TeensyCore)	
USB Type tre Keyboard } els Port	USB Type: "Serial" Keyboard Layout: "US English" Port		•	•	
tre	Programmer: "USBtinyISP"		•		

You can then select which USB device core you want to use



Unlike modern Arduino boards, this board does not auto-reset! Before uploading, you'll need to press the RESET button to get the BOOT LED pulsing

### Download

#### Download

- EagleCAD PCB files on GitHub (http://adafru.it/lf3)
- Fritzing object available in the Adafruit Fritzing Library(http://adafru.it/aP3)
- The LUFA USB-stack website (http://adafru.it/lf5)
- Our minor fork to the LUFA core (http://adafru.it/lf6) this is where our AVR109 bootloader lives (in Bootloaders/CDC)

Download Windows Driver atmega32u4cdc.inf http://adafru.it/lf7

### Schematic



## **Fabrication Print**

Dims in inches

