

## Product: USB2AX



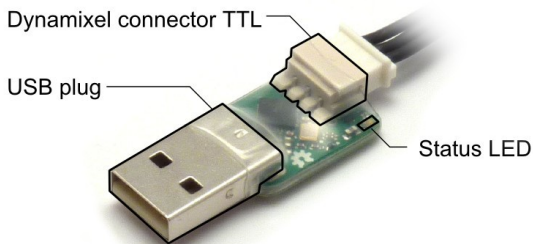
### What is the USB2AX?

The USB2AX is a small interface to control Dynamixel servomotors ([http://www.robotic.com/se/dynamixel\\_en](http://www.robotic.com/se/dynamixel_en)) like the AX-12 from a computer. It plugs into a USB port and has a 3-pin Dynamixel connector to be connected to the servos.

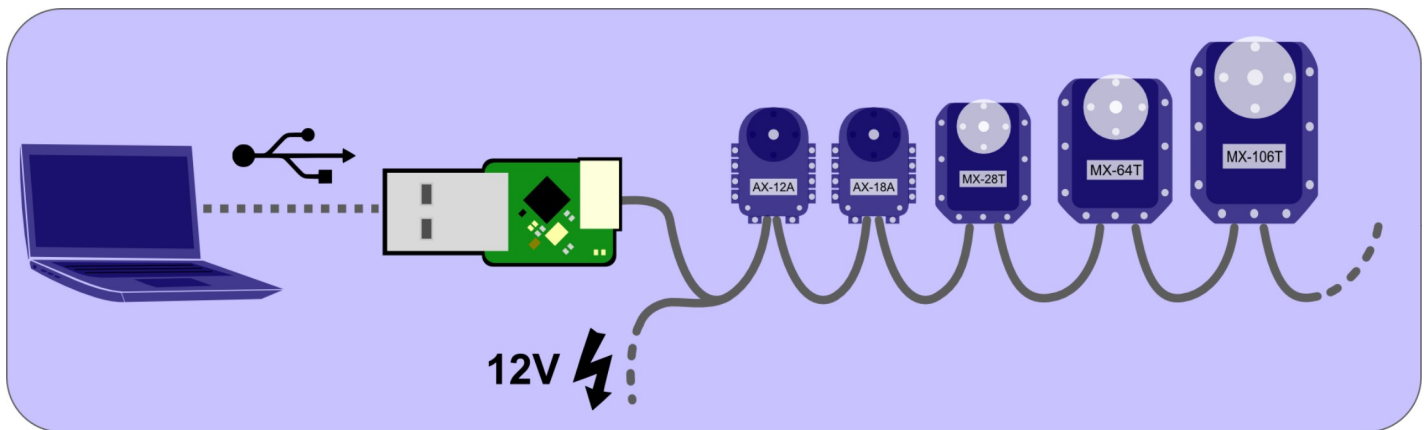
If you're in a hurry, jump to [Quickstart](#) !

**Documentation en Français** sur le Wiki de Cybedroid ([http://wiki.cybedroid.net/index.php?title=Carte\\_adaptateur\\_USB2AX\\_Xevelabs](http://wiki.cybedroid.net/index.php?title=Carte_adaptateur_USB2AX_Xevelabs)) (merci!)

► To fix problems experienced between USB2AX 3.1a and earlier and RoboPlus v1.1x, please [update the firmware](#) ►  
 USB2AX 3.2a already shipped with firmware v04 containing the correction.



Status LED: the LED lights up GREEN when the USB2AX is ready, RED if not (no driver, USB authentication error...).  
 The pads and jumper under the PCB are detailed in the [GPIO and SPI](#) page.



### What is it useful for then?

The USB2AX is a direct way to control servos to move legs, arms or claws in a robot with an embedded computer (laptop or Raspberry Pi <http://www.raspberrypi.org/>) for example.  
 You can also use it as a tool to setup, diagnose or fix your servos.  
 Have a look at some examples of application!

### What servos can it talk to?

The USB2AX can talk with the Dynamixel servos ([http://www.robotic.com/se/dynamixel\\_en](http://www.robotic.com/se/dynamixel_en)) which use a half-duplex TTL interface (3-pins connector).  
 More specifically:

- All of the AX servos (AX-12/AX-12+/AX-12A, AX-12W, AX-18/AX-18A).
- The "T" versions of the MX servos (MX-28T, MX-64T, MX-106T).

### Features

- Communication with AX and MX Dynamixel servos (with 3 wires) @ 1Mbps
- Very small (16x6mm)
- Optimized latency
- Protection against ESD and short circuits
- Advanced functionalities: [sync\\_read...](#)
- Windows / Linux / MacOS
- Compatible with RoboPlus, Dynamixel SDK, ROS, Python, C/C++/C#, Visual Basic, Java, Labview, Matlab...
- Firmware updates
- Open Source Software & Hardware, programmable and hackable!

### What are the limitations?

Fortunately, the USB2AX can not do everything ☹️

- The USB2AX can not communicate with the Dynamixel servos using the RS485 interface (the one with 4 pins).
- Neither can it control any sort of hobby/RC servos or other brand of robot servos.
- Compared to the USB2Dynamixel, it also lacks the RS232 port (with its big DB9 connector) used to program some controllers from Robotis like the CM-510.
- It can not supply power to the servos (see [Q: How to power the servos](#)).
- It is not a generic USB to Serial (even though it can be modified to be one) and has very different capabilities than the LN-101.
- The USB2AX reserves the ID 0x00, so it cannot be used by a servo on the bus.
- It does not make coffee, but could be used to create a robot that does!

### Quickstart

Read the [Quickstart Guide](#), and be on your way ☺️

For Linux users, have a look at the FAQ ([http://www.xevelabs.com/doku.php?id=product:usb2ax:faq#dynamixel\\_sdkhow\\_do\\_i\\_use\\_it\\_with\\_the\\_usb2ax](http://www.xevelabs.com/doku.php?id=product:usb2ax:faq#dynamixel_sdkhow_do_i_use_it_with_the_usb2ax)) too for some additional "gotchas".

To go further...

More information for everyone:

- [FAQ](#)
- [Detailed specifications](#)
- [How to power the servos?](#)

### Some more stuff for power-users:

- [Advanced instructions: sync, read, etc...](#)
- [How To update the firmware?](#)
- [A tentative guide on achieving low latency with the USB2AX](#)
- [Dynamixel protocol additional information](#)
- [Troubleshooting your Dynamixel setup](#)

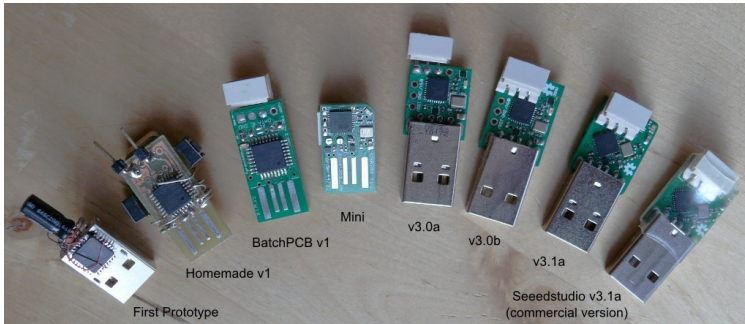
Advanced topics to use the full potential of the hardware for your specific application

- Customize the firmware
- GPIO and SPI
- Unbricking and programming using ICSP

### Miscellaneous

TODO list for the USB2AX...

## History



First prototypes [<http://www.xevel.fr/blog/index.php?post/2011/11/24/USB2AX%3A-the-early-days..>], v1.0 [<http://www.xevel.fr/blog/index.php?post/2011/08/31/USB2AX>], v3.0a

<http://www.svef.se/blog/index.php/post/2012/12/06/The-shape-of-things-to-come-v3.1b/> <http://nforum.vy.com/viewtopic.php?t=448&start=33>, USR2AX Mini <http://www.svef.se/blog/index.php/post/2012/04/03/USR2AX-min1-v3.1a/> <http://forums.crosstonsbotics.com/download.php?575:USR2AX-v3.1>.  
v2 never saw the light of day, it was very similar to v1 and scrapped when I realized it was going to be much bigger than v1. Instead, I went on with v3 that used smaller components (0402 passives instead of 0603, QFN instead of QFP for the ATmega). And finally the commercial v3.1a, which differs from the original v3.1a only by its corners: they were rounded, they are now square for manufacturing reasons.

v3.2a that went on sale on Mai 2014 is a slightly more rationalized version of the 3.1a, which uses more of the Seedstudio Open Part Library components in place of more expensive and difficult to source ones (crystal and LEDs mostly)

### Examples of projects using USB2AX

Xachikoma by Xevel [[http://forums.trossenrobotics.com/robots.php?project\\_id=8](http://forums.trossenrobotics.com/robots.php?project_id=8)], the four-legged reason for creating the USB2AX. 4 legs, 5DOF each, plus powered wheels instead of feet.

Charlotte by KevinO [<http://forums.trossenrobotics.com/showthread.php?p040-PhantomX-controlled-by-a-Raspberry-Pi&p=56474#post56474>], an hexapod controlled with a Raspberry Pi through an USB2AX

HexaWheels by Siempre Aprendiendo [<https://www.youtube.com/watch?v=Yhv43H5Omfc>], an hexapod with wheels and a Raspberry Pi.

Golem by KevinO [<http://www.youtube.com/watch?v=jAEUkANYaAI>], a big hexapod using an Intel NUC.

demo with BeagleBone Black by bloffin3000 [http://www.youtube.com/watch?v=O\_EMc5umdDw]

USB2AX on the web

Xevelabs [<http://www.xevel.fr/blog/index.php?category/Xevelabs/USB2AX>], Xevel's blog

PoBot (<http://www.pobot.org/USB2AX-une-interface-USB-pour-bus.html>) (FR), a review by the French robotics associ-

#### Acknowledgements

### Contributors

## Julien Le Guec

Richard Ibbotson (IBot) - optimization of the code for lower latency, improvements to the schematics, implementation of ad

Dean Camera for making the greatest USB library for AVR, LUFA (<http://www.fourwalledcubed.com>)

Weston T. Schmidt for making the cross-platform app DFU-Programmer [<http://sourceforge.net/projects/dfu-programmer/>]  
 Gabor Molnár [[http://wiki.gabor-molnar.net/index.php/ArduinoGame\\_adapters/USB2AV](http://wiki.gabor-molnar.net/index.php/ArduinoGame_adapters/USB2AV)] for his demonstration on Emu68k / for documentation in

Trossen Robotics [<http://www.trossenrobotics.com/>], Robosavvy [<http://robosavvy.com>], SeedStudio [<http://www.seedstudio.com>] and Adafruit [<http://adafruit.com/>] for the help (direct or indirect) to bring the USB2AX to market.

Inspirations:

Arduino Uno [<http://arduino.cc/en/Main/ArduinoBoardUno>]

Ruggeduino (<http://ruggedcircuits.com/html/ruggeduino.html>)

Download / files

- Schematics and

- Latest firmware
- STL 3D model [[http://www.xevel.fr/other/usb2ax\\_v3.1.stl](http://www.xevel.fr/other/usb2ax_v3.1.stl)] of the USB2AX v3.1a board
- File for a 3D printed case 