

Arduino Playground is read-only starting December 31st, 2018. For more info please look at this [Forum Post](https://forum.arduino.cc/index.php?topic=580991.0)
(<https://forum.arduino.cc/index.php?topic=580991.0>).

Manuals and Curriculum
(../ManualsAndCurriculum/)
Arduino StackExchange
(<https://arduino.stackexchange.com/>)
Board Setup and Configuration
(../ArduinoCoreHardware/)
Development Tools
(../DevelopmentTools/)
Arduino on other Chips
(../ArduinoOnOtherAtmelChips/)
Interfacing With Hardware
(../InterfacingWithHardware/)

- Output
(../InterfacingWithHardware/#Output)
- Input
(../InterfacingWithHardware/#InputTOC)
- User Interface
(../InterfacingWithHardware/#ui)
- Storage
(../InterfacingWithHardware/#Storage)
- Communication
(../InterfacingWithHardware/#Commun)
- Power supplies (../IntWithHW-PwrSup/)
- General
(../InterfacingWithHardware/#General)

Interfacing with Software
(../InterfacingWithSoftware/)

User Code Library
(../GeneralCodeLibrary/)

- Snippets and Sketches
(../SketchList/)
- Libraries (../LibraryList/)
- Tutorials (../TutorialList/)

Suggestions & Bugs
(<https://github.com/arduino/arduino/issues>)

Electronics Technique
(../ElectroInfoResources/)

Sources for Electronic Parts
(../Resources/)

Related Hardware and Initiatives
(../SimilarBoards/)

Arduino People/Groups & Sites
(../People/)

Exhibition
(../Projects/ArduinoUsers/)

Project Ideas (../Projects/Ideas/)

Languages (../Languages/)

Participate (../Participate/)

- Formatting guidelines
(../Participate/#contribrules)
- All recent changes
(../Site/AllRecentChanges/)
- PmWiki (../PmWiki/PmWiki/)
- WikiSandBox training
(../WikiSandbox/)
- Basic Editing
(../PmWiki/BasicEditing/)
- Documentation index
(<https://www.pmwiki.org/wiki/PmWiki/DocumentationIndex>)

General

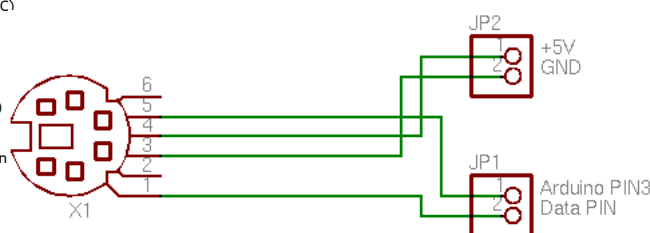
The PS2Keyboard library uses one of the two available external interrupts to react on keyboard input. Once such input is received, it's stored in a one-byte buffer and is available to be read.

Latest Version

Version 2.3 (https://www.pjrc.com/teensy/td_libs_PS2Keyboard.html) - Minor bugs fixed.

Connection

The following schematic shows how to connect a PS/2 connector:



Make sure you connect the **Clock PIN** to the digital PIN 3 on the Arduino. Otherwise the interrupt, and therefore the whole library, won't work.

Usage

Below is a simple example on how to use the library. For more keycode constants have a look at the `PS2Keyboard.h` file.

```
#include <PS2Keyboard.h>
#define DATA_PIN 4
PS2Keyboard keyboard;

void setup() {
  keyboard.begin(DATA_PIN);

  Serial.begin(9600);
  Serial.println("hi");
  delay(1000);
}

void loop() {
  if(keyboard.available()) {
    byte dat = keyboard.read();
    byte val = dat - '0';

    if(val >= 0 && val <= 9) {
      Serial.print(val, DEC);
    } else if(dat == PS2_KC_ENTER) {
      Serial.println();
    } else if(dat == PS2_KC_ESC) {
      Serial.println("[ESC]");
    }
  }
}
```

Version Histoy

002

Attach:PS2Keyboard002.zip (../uploads/Main/PS2Keyboard002/index.zip)

This library is at a very early stage. If anyone wants to revise/rewrite it, feel free to do so (published under the LGPL).

014A

Help(<https://support.arduino.cc>)

Attach:PS2Keyboard_014A.zip (../uploads/Main/PS2Keyboard_014A/index.zip)

2.0

http://www.pjrc.com/teensy/td_libs_PS2Keyboard.html
(https://www.pjrc.com/teensy/td_libs_PS2Keyboard.html)

EDITED AGAIN - Actually mostly rewritten

Version 2.0 adds the following improvements:

- Scan Code Buffering: Data captured by the interrupt is stored into a circular buffer, so several keystrokes can be buffered if your sketch is busy and can not call read() between each one. Buffering also retains key break (up) events. Previous versions buffered only a single scan code and discarded all key break codes.
- Shift Key State: The state of both shift keys is tracked, so conversion to ASCII produces proper upper and lower case characters. The numbers and other dual character keys also properly respond to shift. Previous versions were limited to only uppercase and non-shift output.
- ASCII Only Output: The read() function returns only ASCII characters, or special keys mapped to ASCII control characters. Previous versions returned a mix of ASCII and raw scan codes, with no way to distinguish between them.
- Efficient Conversion: Fast table lookup has replaced a slow linear search algorithm.
- Multi-Board Support: Arduino Mega, Teensy and Sanguino interrupt pins are supported. Other boards can be easily added, or will work automatically if their core files define certain aliases.

Some of the information on this page is now obsolete. Please refer to the version 2.0 page for up-to-date documentation.

2.1

http://www.pjrc.com/teensy/td_libs_PS2Keyboard.html
(https://www.pjrc.com/teensy/td_libs_PS2Keyboard.html)

Version 2.1 adds the following improvements:

- Automatically resync bitstream after a 1/4 second timeout.
- Compatible with Arduino "new-extension" branch.

2.3-Ctrl

Download Attach:PS2Keyboard_2.3-Ctrl.zip (../..../uploads/Main/PS2Keyboard_2.3-Ctrl/index.zip)

CTRL SUPPORT

NOTE: Based on v2.3 from Teensy web site with small addition to the main code to recognize Ctrl modifier.

When Ctrl is held, alphabetic characters (A, B, C, ...) are substituted with low ASCII range (codes 1, 2, 3, ...). It is case insensitive. The motivation is embedded projects with mode changes or keyboard commands.

The examples folder contains Ctrl_Test sketch with a demonstration. The code would look something like this:

```
// Ctrl + <alphabet key> produces ASCII codes 1-26
} else if (c < 26) {
  Serial.print("^");
  Serial.print((char)('A' + c - 1));
```

2.3-Ctrl-Enter

Download Attach:PS2Keyboard_2.3-Ctrl-Enter.zip (../..../uploads/Main/PS2Keyboard_2.3-Ctrl-Enter/index.zip)

CTRL+ENTER SUPPORT

Help(<https://support.arduino.cc>)

- Moved PS2_DOWNARROW to code 12
- Example(Ctrl_Test.ino)

```
// check for some of the special keys
if (c == PS2_ENTER) {
    Serial.println();
} else if (c == PS2_LINEFEED) {
    Serial.print("[^Enter]");
}
```

Further readings

- <http://www.beyondlogic.org/keyboard/keybrd.htm>
(<https://www.beyondlogic.org/keyboard/keybrd.htm>)
- <http://www.computer-engineering.org/ps2keyboard/scancodes2.html> (<https://www.computer-engineering.org/ps2keyboard/scancodes2.html>)
- BarcodeScanner (../ComponentLib/BarcodeScanner/)
- PS2KeyboardExt (../PS2KeyboardExt/)
- <https://web-beta.archive.org/web/20130421110617/http://nocommercialpotential.posterous.com/interfacing-the-ps2-keyboard> (<https://web-beta.archive.org/web/20130421110617/https://nocommercialpotential.posterous.com/interfacing-the-ps2-keyboard>)