

---

# *USB2PPM*

## User's Guide

---

Version 2.00  
dated 09/23/15

Gregor Schlechtriem

## Content

<b>Overview</b>	<b>3</b>
<b>PCC PiKoder Control Center</b>	<b>5</b>
Getting started.....	5
Real-time Control.....	7
PPM Settings.....	7
<b>Flytron CompuFly USBtoPPM Joystick Interface</b>	<b>9</b>
Getting started.....	9
Real-time Control.....	10
<b>Serial Interface</b>	<b>11</b>
ASCII Command Interface .....	11
CompuFly USB PPM converter interface.....	12
<b>Pin Description and Packaging</b>	<b>15</b>
Description of pins .....	16
<b>Standard application</b>	<b>17</b>
LED status indication .....	18
<b>CompuFly USB PPM converter</b>	<b>19</b>

## **Overview**

The USB2PPM (PiKoder/USB) is a true single chip solution using a modern PIC controller to generate an eight channel PPM-Stream with a resolution of 1  $\mu$ s. This stream can be used by many R/C transmitters for flying with a joystick a mouse, etc. by simply connecting the PiKoder output to the trainer connector of the R/C transmitter. The power for the USB2PPM converter is usually provided through the computer's USB port.

The USB2PPM supports two control protocols: the “CompuFly USB PPM converter”-protocol, which allows you to use the “CompuFly Joystick Interface”. Also the PiKoder features a two-way ASCII-Protocol designed to be used in combination with standard terminal programs such as (but not limited to) Tera Term and TTY.

A free graphical and intuitive configuration and control program for Windows 7, the “PCC (PiKoder Control Center)” is available for download at [www.pikoder.com](http://www.pikoder.com), making it simple to test and program the controller.

This User's Guide is based on the most recent hard- and firmware available for the USB2PPM and the related PCC programming software. Please check for updated information and new software releases on [www.pikoder.com](http://www.pikoder.com).

Please share with me any comments, improvement ideas or errors you will find or encounter in working with your PiKoder. I can be reached at [webmaster@pikoder.com](mailto:webmaster@pikoder.com). Thank you very much!



---

# 2

---

## ***PCC PiKoder Control Center***

The USB2PPM's serial interface provides access to configuration options as well as support for real time control. The PCC PiKoder Control Center is a graphical tool that makes it easy for you to use this interface. For almost any project you will start by using the PCC PiKoder Control Center to set up and test your PiKoder. This section explains the features of the PCC PiKoder Control Center.

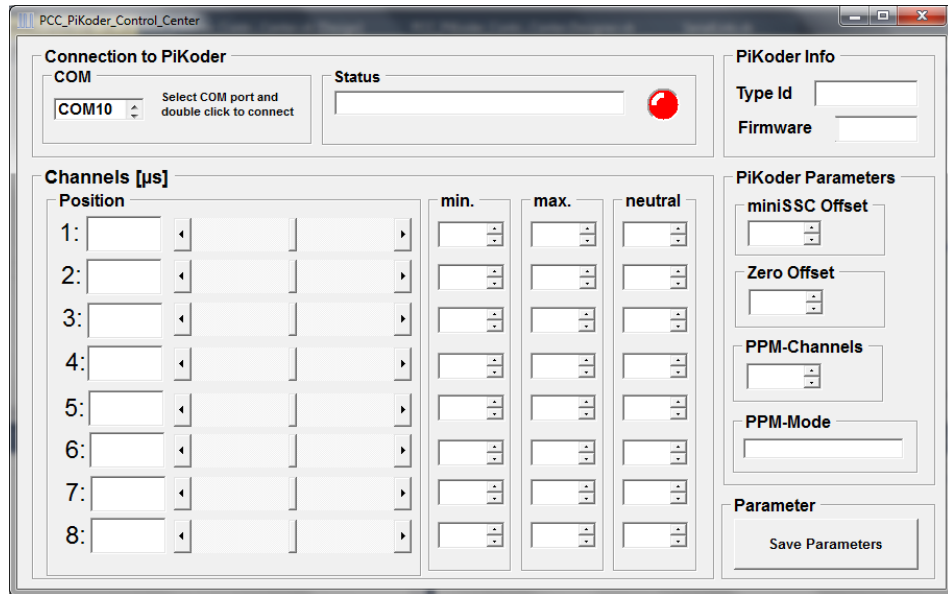
### **Getting started**

The hardware setup for the interface is simple and straight forward: Please refer to exhibit B for the standard setup. You have to connect your USB2PPM with the USB port of your PC using a suitable cable. This cable will provide also for the power supply of the PiKoder.

The communication requires the installation of a USB driver, which would map the USB port into a virtual COM port. After connecting the PiKoder/USB to a USB port the driver would be installed following the automated Windows 7 standard procedure: searching the driver on Windows Update, downloading and installing it. If you have chosen the setup according to the standard application (please refer to Appendix B for more information), then LED 2 would be blinking in fast mode during the installation process; once the installation is completed both LEDs would blink alternating and thus indicating that the PiKoder/USB would be ready for programming.

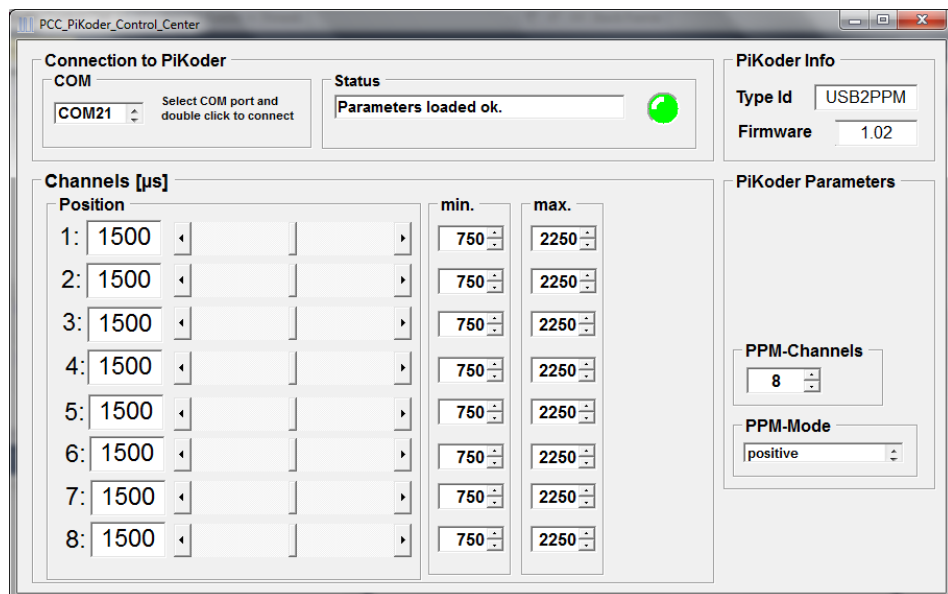
It is highly recommended that you download the latest version of the PCC PiKoder Control Center to enjoy the complete feature set of your PiKoder. The PCC PiKoder Control Center software is Open Source and released under a [Apache License Version 2.0](#). The complete source is available through [github](#).

If you downloaded the executable then please unzip the downloaded file. The program does not require any installation. After starting it you will see the window as shown on the following page.



The PCC PiKoder Control Center application would show COM10 to be the first available port in your computer. Select the COM port your USB2PPM is connected to (in this example COM23, see below) and then click on the box to indicate your selection and communication to the controller will be established.

The LED color would change to green for an online indication and the current parameters and settings of the USB2PPM would be loaded and displayed as shown on the following page. The firmware version of your PiKoder would be displayed in the respective field.



Please note that the layout of the form is adapted to the parameters and features available for the PiKoder firmware version found.

You would now have full control of your PiKoder: either for real-time control by the sliders or for changing the settings with respect to PPM polarity and number of channels.

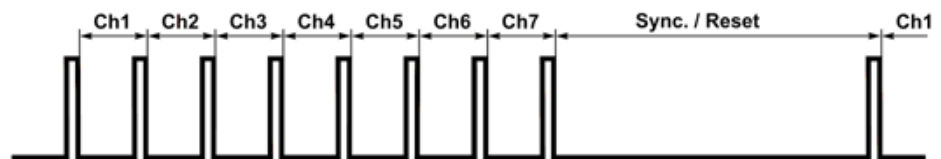
## Real-time Control

The sliders are used for controlling the USB2PPM's outputs and the respective numerical fields monitor the status in real time displaying the current channel value in  $\mu$ s. A separate row of controls is displayed for each of the PiKoder's channels.

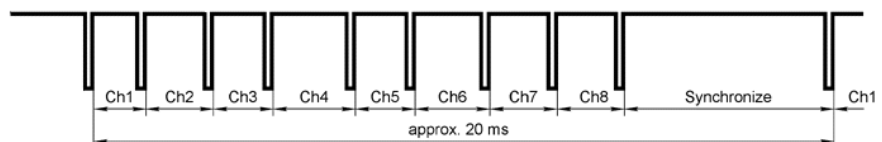
The key parameters for each channel such as min. and max. pulse width and neutral position can be set individually in the respective row. The PCC PiKoder Control Center will limit the slider value to the min. and max. value shown. This feature however is implemented in the Control Center. The PiKoder itself does not perform a parameter check and would therefore accept channel values outside of the shown boundaries. Also, the PiKoder/USB does currently not allow for storing parameters in the controllers non-volatile memory.

## PPM Settings

The PiKoder/USB features either a negative or positive PPM stream. The polarity is related to the output of the PPM\_OUT pin. The following picture presents the signals (please refer to exhibit A for more information). Also, you can select the number of channels encoded in the stream. A range from 1 to 8 channels is supported; more than 8 channels can technically not be encoded in a standard 50 Hz PPM frame. Please note that the PPM setting have no effect if you are using the demo firmware.



Above: “positive” PPM Frame at PPM\_OUT (7 channels selected)



Futaba R/C transmitter trainer interface signal (inverted PPM modulation)  
Ch1..8 - variable pulse width from 1 to 2 ms (center 1,52 ms)  
Synchronize - usually around 5 ms

Above: “negative” PPM Frame at PPM\_OUT (8 channels selected)

Room for your notes:



## ***Flytron CompuFly USBtoPPM Joystick Interface***

The USB2PPM does support the “Flytron CompuFly USBtoPPM Joystick Interface” which is available as open source and can be downloaded on several web pages.

### **Getting started**

The hardware setup for the interface is simple and straight forward: Please refer to exhibit C for the minimal setup. You have to connect your USB2PPM with the USB port of your PC using a suitable cable. This cable will provide also for the power supply of the PiKoder.

The communication requires the installation of a USB driver, which would map the USB port into a virtual COM port. After connecting the PiKoder/USB to a USB port the driver would be installed following the automated Windows 7 standard procedure: searching the driver on Windows Update, downloading and installing it.

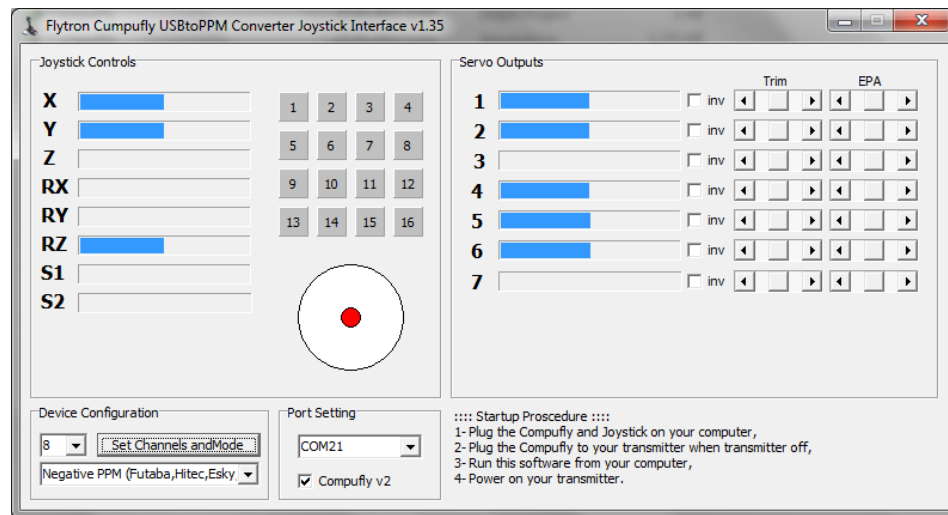
Then you would have to download and install the Joystick interface following the steps indicated. After the installation, start the program and select the COM port which is connected to your PiKoder – in the screen print on the next page this is COM21. You also want to make sure that the “CompuFly v2”-box is checked (which is the default configuration). Now you should be able to control your PiKoder with a joystick.

## Real-time Control

Your servos are controlled through the joystick attached to your computer. Other parameters can be checked by checking the appropriate boxes or moving the sliders.

Please note that more than eight channels are not supported by the USB2PPM since it does not make any sense given the 50 Hz repetition rate of a standard ppm frame.

Also, changing the mode (polarity of the ppm stream) and a change in the number of channels will be executed right away and be effective immediately - a reboot of the converter is not required in the case of the PiKoder.



Please refer to section 4 for more details on the protocol.

---

# 4

---

## ***Serial Interface***

The PiKoder's serial interface is based on a virtual COM port which is managed by the Windows driver. The parameters for the serial transmission are 9600 Baud, 8 data bits, one stop bit, with no parity.

The bytes sent to the USB2PPM are commands which allow you to control the program and control the PiKoder. The USB2PPM supports a two-way ASCII-Protocol named Ascii Command Interface (ACI) designed to support controlling the PiKoder with standard terminal programs such as (but not limited to) TeraTerm, Putty, hyperterm

### **ASCII Command Interface**

The ASCII Command Interface (ACI) is probably the most versatile way to program the PiKoder without any specific host software such as the "PCC PiKoder Control Center". All commands are simple ASCII and are sent using a Windows based terminal program such as Hyperterm or Tera Term. The commands can be typed in right away and the response of the controller is readable without referring to any specific code tables. Please note that neither 'CR' nor 'LF' is needed to send the command to the controller.

There are two basic types of commands: commands for quering parameters and for setting parameters.

If a parameter is read the PiKoder will provide for proper formatting by sending a "CRLF" prior to sending the parameter value and support readability by sending another "CRLF" after the parameter value.

If a parameter is set the PiKoder will acknowledge the proper execution by sending an "!" framed by "CRLF".

If a command could not be interpreted at all then a question mark '?' framed by 'CR' 'LF' would be echoed. Please note that protocol syntax checking is very limited at this point in time.

The following ACI commands are available:

- '?': query the PiKoder type information; PiKoder/USB will respond in a format 'T=USB2PPM' framed by 'CR' 'LF'
- '0': query the firmware version; PiKoder will respond in a format 'n.nn' framed by 'CR' 'LF'
- 'i?': query the current pulse width for channel i ( $i = 1..8$ ); PiKoder will respond 'CR' 'LF' 'xxxx' 'CR' 'LF' with xxxx representing the pulse width in  $\mu\text{s}$
- 'i=xxxx': set the pulse width for channel i to xxxx  $\mu\text{s}$  (xxxx in decimal format,  $i = 1..8$ ); PiKoder will acknowledge execution of the program by sending an 'CR' 'LF' '!' 'CR' 'LF'

## CompuFly USB PPM converter interface

The PiKoder/USB supports the “CompuFly v2” protocol. The hardware required is minimal due to controller chip with an integrated USB port. Please refer to exhibit C for more information.

The USB port implementation of the PiKoder/USB will also take care of the port parameters required for the communication.

The “CompuFly v2” command interface is comprised of three messages:

- 'S' i xxxx: set channel value for channel i (1..7) to 'xxxx'
- 'S' 0x15 0x00 i: set the number of channels for the ppm frame to i (default = 8)
- 'S' 0x16 0x00 i: set the polarity of the ppm signal (0 = positive value (default), 1 = negative value)

Please note the following:

- A ppm frame with more than eight channels does not allow for a repetition rate of 50 Hz and is thus not supported. If the user would try to set the number of channels for the frame to a number  $> 8$  then this request will result in a frame with eightchannels.

- **Room for notes** -



---

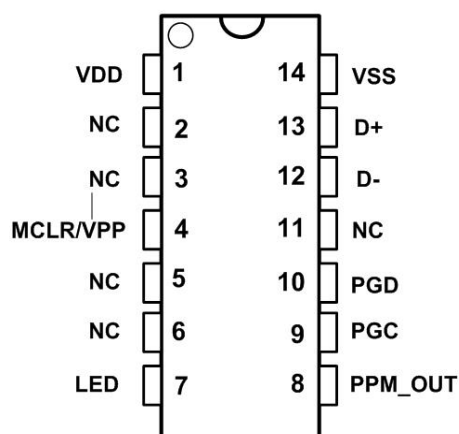
# A

---

## *Pin Description and Packaging*

The PiKoder/USB comes in an 14 pin DIP package (see below). The device operates from 3.3 – 5 Volts. Please refer to the PIC 16F1455 data sheet from Microchip ([www.microchip.com](http://www.microchip.com)) for complete electrical and physical specifications.

A complete description of the pins is given on the following page. If a different package would be required for your application then please contact [sales@pikoder.com](mailto:sales@pikoder.com) for more information.



## Description of pins

Pin	Symbol	Description
1	VDD	Supply voltage. Connect to 3,3 – 5 V DC
2	NC	Not connected (reserved for later use)
3	NC	Not connected (reserved for later use)
4	MCLR/VPP	Reset pin, active low. Connects directly to Vss for automatic reset at power up.
5	NC	Not connected (reserved for later use)
6	NC	Not connected (reserved for later use)
7	LED	LED indicator output (please refer to Appendix B for more information)
8	PPM_OUT	Output pin with PPM signal
9	PGC	Clock pin for In-Circuit-Serial-Programming
10	PGD	Data pin for In-Circuit-Serial-Programming
11	NC	Not connected (reserved for later use)
12	D-	USB data line
13	D+	USD data line
14	VSS	Ground connection



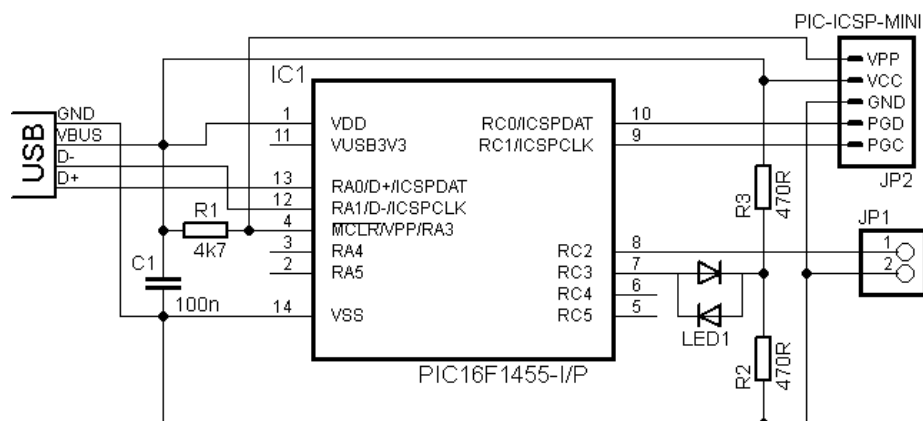
---

# B

---

## *Standard application*

The following schematic shows the standard application of the USB2PPM. Please note that the bi-color LED indicates the USB status and is optional.



The PPM\_OUT signal is available at PIN RC2 which is connected to PIN 1 of JP1 in the above schematic.

## LED status indication

This following table shows the status of the USB connection (“DEVICE STATE”) based on the LED signaling state (“LED RESPONSE”).

DEVICE STATE	LED RESPONSE
DETACHED	OFF
ATTACHED	RED
POWERED	RED
DEFAULT	YELLOW
ADRESS	YELLOW
CONFIGURED	GREEN

---

# C

---

## *CompuFly USB PPM converter*

The following schematic shows the hardware setup for the CompuFly PPM converter which does not require any indicator LEDs or in-circuit programming and is therefore even less complex than the standard application.

