

KBD01

Keyboard Interface for Encoders and Switches

Copyright 2015 Detlef Mahr

Rev 1.2

Contents

1. General	1
2. Rotary Encoders	1
3. Switches	2
4. The Switch Matrix	2
5. HIDMacros	2
6. Configuration	3
7. Configuration Program	3
8. Encoders 1 ... 8 / Switches 1 ... 16	5
9. Switches 17 ... 32	6
10. LEDs	6
11. Switching Matrix	7
12. Interface Layout	7
13. Pin Descriptions	8
14. Wiring Examples	8
15. Hints	9

1. General

The keyboard interface KBD01 maps switch actions to key presses - a switch which is closed or opened sends a keystroke to the PC, freely selectable from the characters on a standard keyboard, optional with the Shift, Ctrl, and Alt keys combined. The length of the keypress is usually 10 ms, but it is adjustable in a wide range (1 to 250 ms). Also the (minimum) break between successive keystrokes is adjustable in this range.

Both, encoders and switches can be connected to the interface. Up to 64 different switching events can be associated with selectable key combinations. With the encoders, these are a sequence of key presses when turning clockwise and a different one when turning counter-clockwise. With the switches, the *ON* and *OFF* events are issued separately. All switching actions are software debounced, that is to say, even inexpensive switches can be used reliably.

It is possible, to connect up to 8 encoders with push-button switch (or separate switch) and additional 16 pushbutton or toggle switches. Inputs not used for the rotary encoders can be used for switches.

2. Rotary Encoders

Mechanical rotary encoders usually have three terminals, a central (*C*), which cyclically contact the second (*A*) and the third port (*B*) when spinning. Through constructive skillful change of *contact* and *non-contact* the direction of rotation can be found and a separate keystroke for the clockwise and counter-clockwise rotation is issued. Dual encoders with concentric axis, logically have three additional connections (*C'*, *A'* und *B'*), which work independent of the first - like two separate encoders. A push button is often installed as well, with two additional connections.

Optical encoders also need a power supply (usually 5 volts) and have commonly 4 connections: *A* and *B* (as with the mechanical counterparts) as well as *+5V* and *GND* (no *C* terminal). The wiring can vary from type to type and must be taken from the respective data sheet.

An encoder input on the KBD01 has four ports, three for *C*, *A* and *B*, plus one for a pushbutton or separate switch (*D*). In the case of a dual encoder a second input for *C'*, *A'* und *B'* must be used.

Each encoder can be virtually doubled in its function, which means that an additional pair of clockwise and counter-clockwise keycodes is available. The switchover then happens by the pushbutton with the appropriate configuration of the KBD01.

If a high-resolution encoder is cranked faster as the keypresses can follow, an overrun feature can be turned on, to continue the still missing key presses even after end of the rotation. This is with short keystrokes (~10ms) hardly noticeable. With longer keystrokes it is recommended, however, to turn the overrun off. Then keystrokes of the given length will be issued only as long as the encoder is moved, regardless how many pulses the shaft encoder has actually made.

3. Switches

Simple switches have two connectors. Depending on the position of the switch they are electrically separated or shorted (*OFF* - *ON*). Any change from *OFF* to *ON* and from *ON* to *OFF* is associated with a virtual press of a key. Through the appropriate configuration of the interface this can be restricted to the change from *OFF* to *ON* only, if, for example, a pushbutton shall only trigger a signal when pushed, but not when it is released.

Change-over switches (toggle switches) are in principle two single switches, which are mechanically coupled, such that one switches *ON* when the other switches *OFF* and vice versa. Since these switches have also two positions, there is effectively no difference to the simple switch.

A switch with center position can be seen as two separate switches. However, they cannot be turned on at the same time, but turned off. The three positions can be allocated to three different keystrokes, if for the respective transition from *ON* to *OFF* the *same* key combination is selected. Otherwise you have two different keystrokes for the center position, depending on which direction the switch is placed in the center position.

Multi-switches, mostly rotary switches, can be used also. However, the special features of the switch matrix is to take into account (see *Technical Appendix*).

4. Switch matrix

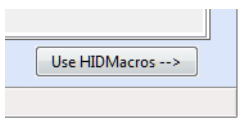
To bring in line the large number of connections with the lower number of microprocessor pins, a switch matrix is used. That is, the switches are arranged at the intersections of a grid formed from 16 rows by 3 columns. This means on the other hand, only those switch terminals are electrically connected together, which are wired to the same row or column line. Therefore it is to make sure that no shorts between individual rows or columns are made in the wiring. Usually, this is not a problem, if every switch and each encoder is connected with the appropriate terminals on the interface. It is a different with multiple switches where mostly a common pole for multiple switch positions is provided. This pole then has to be wired to one of the three column lines and the others to the corresponding row lines (see example in the *Technical Appendix*).

5. HIDMacros

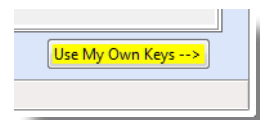
Each of several keyboards connected to a PC is equivalent, that is to say, it makes no difference whether the F6 key is pressed on one keyboard or the other, the effect is the same. This applies to the KBD01 also. Shortcuts already assigned for the default keyboard from a program can be run by the interface as well, but just not as an extension to the standard keyboard but in parallel. A second and a third interface makes no difference.

Different with the very good program *HIDMacros* of Petr Medek (www.hidmacros.eu). Here, each of the connected keyboards is independent of all other seen. For example, if the F6 key on the standard keyboard triggers event *A*, the F6 key on the KBD01 may trigger an event *B* and on a further interface starts action *C*, etc.

The KBD01 usually assigns each switch event a key, being modifiable with the *Shift*, *Ctrl*, and *Alt* keys. *HIDMacros*, however, only accepts unmodified keycodes, but at the time of transfer it can be modified with *Shift*,



The *HIDMacros* feature is enabled by clicking on the button in the bottom right corner. It turns yellow. To use your own key definitions you click the button again.



Ctrl and *Alt* (or even more complex macros, see there). When using *HIDMacros* the KBD01 can be switched to a simple mode where each switch input maps to a fixed key, without the need for an individual configuration. A possibly already existing configuration is retained and can be re-enabled at any time.).

AutoHotkey (www.autohotkey.com/) is another very useful program to combine keystrokes with a powerful scripting language.

6. Configuration

The configuration, so the assignment of a shortcut to a switching event, and the way in which the switches are processed, is stored in the KBD01 and will be preserved even without power supply. A change is done via an easy-to-use program which automatically recognizes all of the connected interface cards. The existing configuration can be read, changed and written back. A configuration can be archived in a file and read back again if necessary. The archive file is in XML-format, and can therefore be read and edited with a simple text editor.

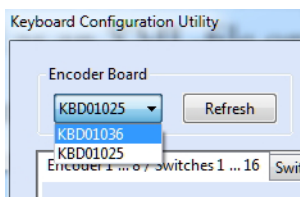
The mapping for the *HIDMacros* mode is permanently stored in the KBD01 and can not be altered.

7. Configuration Program

The *Keyboard Configuration Utility* program displays the current configuration of each of the KBD01 connected to the USB. Changes can be made easily here and the results are stored back into the device. Each configuration can be also saved as an XML-file on disk and read back from there.

After starting the program it searches for all interface cards connected and lists them. The configuration of the last found will be read and displayed in the program window.

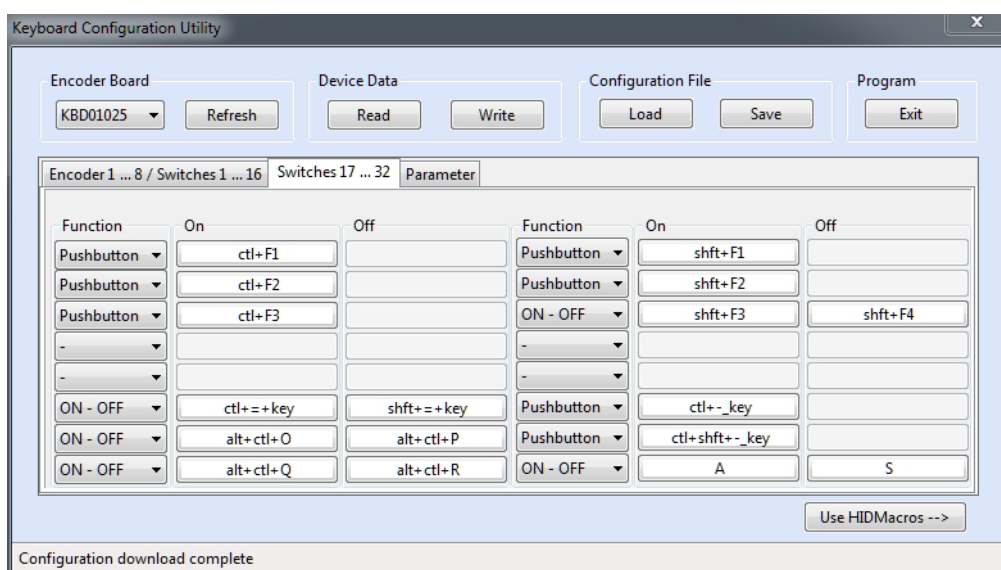
In the **status bar** the number of interface cards found is shown.



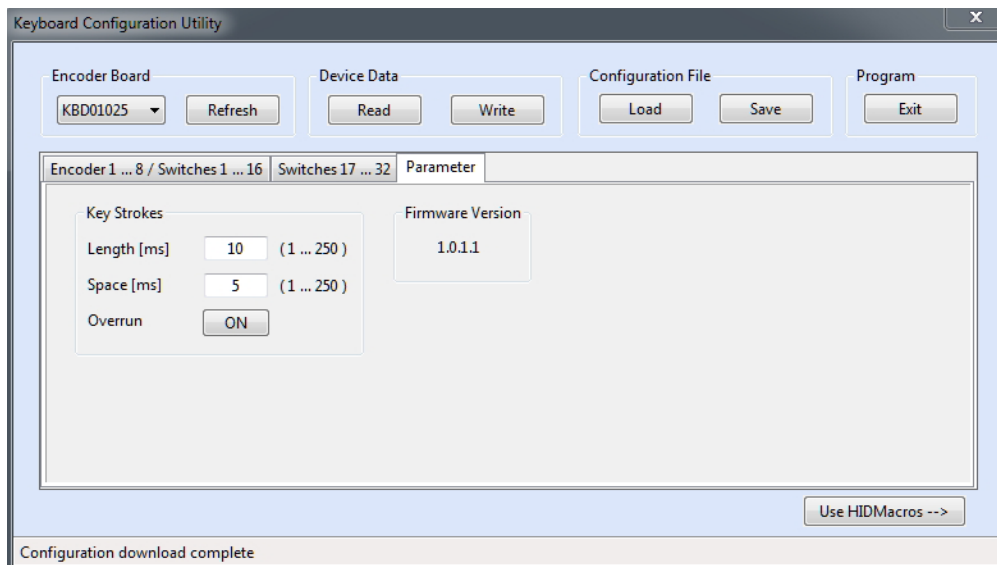
In the selection field on the top left of the program window the desired board can be selected and subsequently edited.

If boards will be removed or new ones attached, the status is updated with the **Refresh** button.

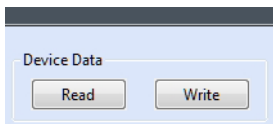
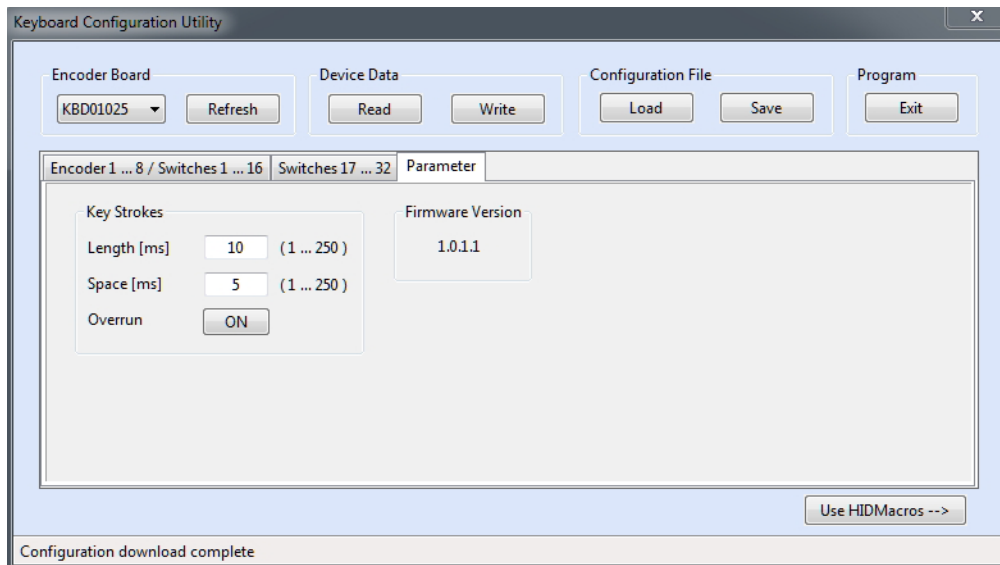
The startup screen shows the configuration of the **encoder inputs**:



On the second page the configuration of the switch inputs is to be seen:



On the third page you will find additional parameters:

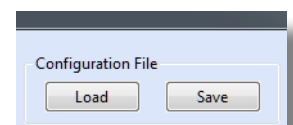


With the **Read** button, the configuration of the selected interface card is re-read and in the appropriate input fields displayed. .

Vice versa, with the button **Write** the currently displayed configuration is written back into the interface.

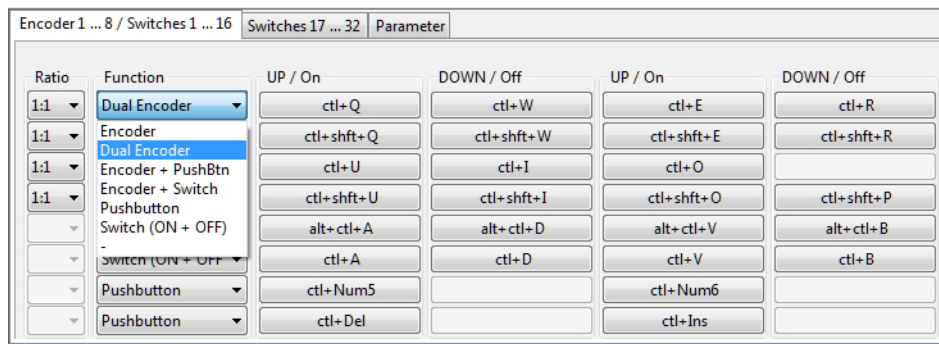
With the **Load** button, a configuration which has been saved on a storage medium can be restored and the input fields will be updated.

Accordingly, with the **Save** button the currently displayed configuration will be stored as an XML file on any storage medium.



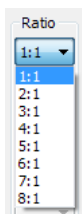
8. Encoders 1 ... 8 / Switches 1 ... 16

This page contains individual settings for the encoder inputs. Encoder inputs can send 2 to 4 keycodes, depending on the configuration:



Options:

Function	UP / On	DOWN / Off	UP / On	DOWN / Off
Encoder A simple encoder is connected	Keycode for UP (clockwise rotation)	Keycode for DOWN (counter-clockwise rotation)		
Dual Encoder An encoder with pushbutton is connected	First keycode for UP (clockwise rotation)	First keycode for DOWN (counter clockwise rotation)	Second keycode for UP (clockwise rotation)	Second keycode for DOWN (counter clockwise rotation)
Encoder + Pushbutton An encoder with pushbutton is connected	Keycode for UP (clockwise rotation)	Keycode for DOWN (counter clockwise rotation)	Keycode for pushbutton ON	
Encoder + Switch An encoder and a switch is connected	Keycode for UP (clockwise rotation)	Keycode for DOWN (counter clockwise rotation)	Keycode for switch ON	Keycode for switch OFF
Pushbutton 1 or 2 Pushbuttons are connected	Keycode for ON pushbutton #1		Keycode for ON pushbutton #2	
Switch (On + Off) 1 or 2 switches are connected	Keycode for ON switch #1	Keycode for OFF switch #1	Keycode for ON switch #2	Keycode for OFF switch #2
- Input is unused				



The input field **Ratio** selects the reduction of the encoder pulses by the given ratio. For example, in a setting of 4:1 each fourth pulse of the encoder leads to a keycode output. Thus high-resolution encoders can be adjusted as necessary, or encoders with more than one (electrical) pulses per dent may be adapted.

Each line represents one encoder input:

This includes 4 configuration fields for the desired key combination. These are the four possible events, UP (turning clockwise) and DOWN (turning anti-clockwise) - or ON and OFF - for single or double encoders or switches. Depending on the selected function, configuration fields not required are grayed and cannot be edited. For example, the function **Rotary Encoder** has only two configuration options, UP and DOWN, the function **Encoder + Switch** but four, UP and DOWN, as well as ON and OFF.

By a single mouse-click on a configuration field - it changes its color to red then - the next key press is captured and displayed. Each key on the keyboard is allowed and you can modify it with the *Shift*, *Ctrl* and *Alt* keys. Normally the displayed key corresponds to an American keyboard: instead of a German „ö“ therefore a „`@key“ appears and instead of a „ß“ a „[key“. I. e. when „ü“ is pressed with all three modifiers, then the configuration field shows „Alt+Ctl+Shft+;key“.

After a click with the mouse (the field is red), but no change is to be made, then a second mouse click aborts this action.

9. Switches 17 ... 32

This page contains all of the settings for pure switch inputs. Switch inputs can send 1 or 2 keystrokes, depending on it's configuration:

Function	On	Off
Pushbutton A switch is connected	keycode for ON	keycode for OFF
On + Off A switch is connected	keycode for ON	keycode for OFF
- Input is unused		

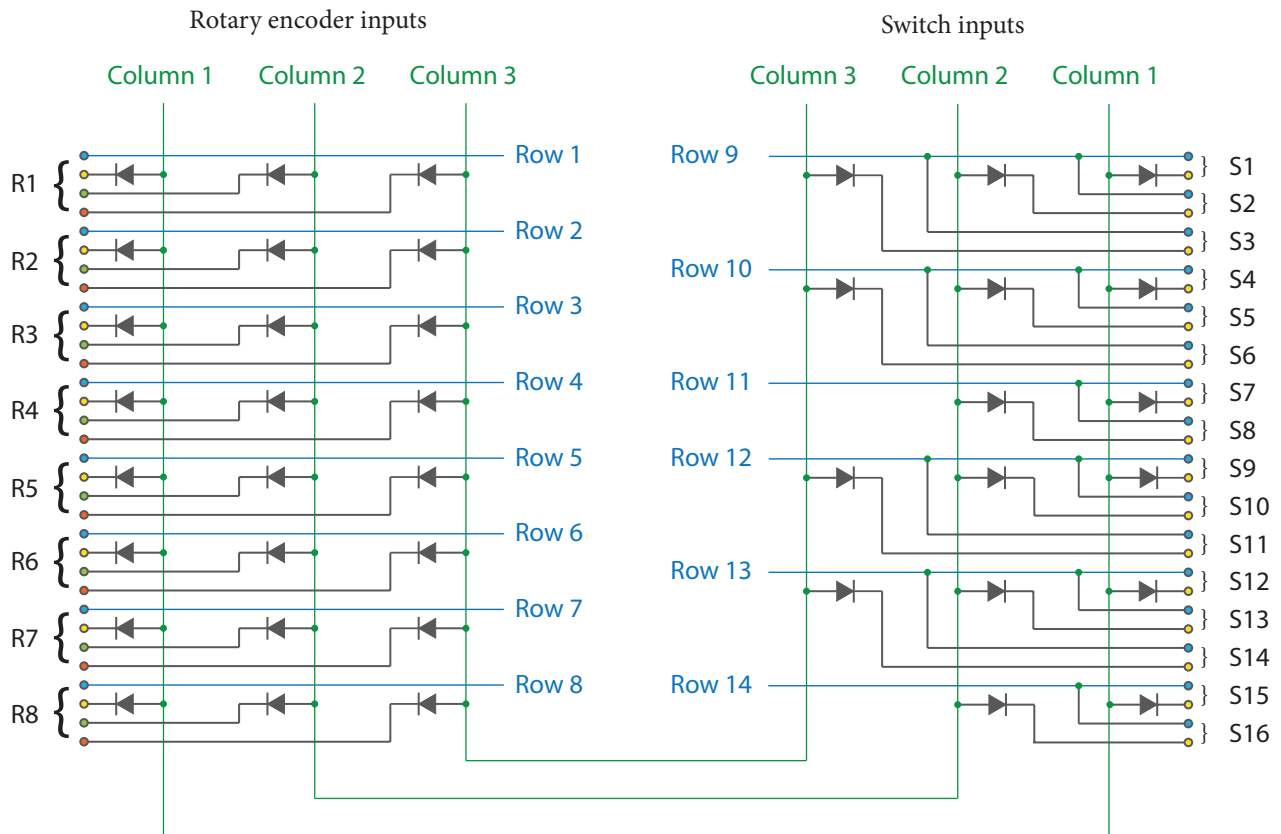
Here the same applies as said above for the configuration fields of the encoder inputs.

LEDs

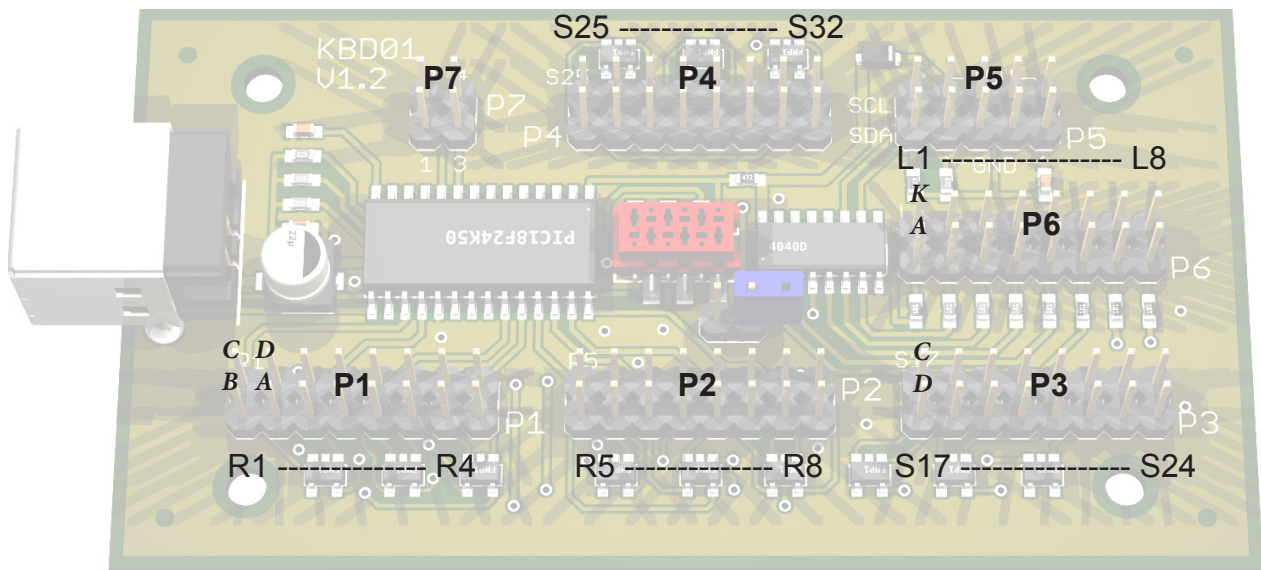
When the function **Dual Encoder** is selected, the eight Leds indicate whether the alternative keycode output is turned on (LED on) or not (LED off).

Technical Appendix

10. Switch Matrix



12. Interface Layout



P.. - Pin header

R.. - Rotary encoder inputs

S.. - Switch inputs

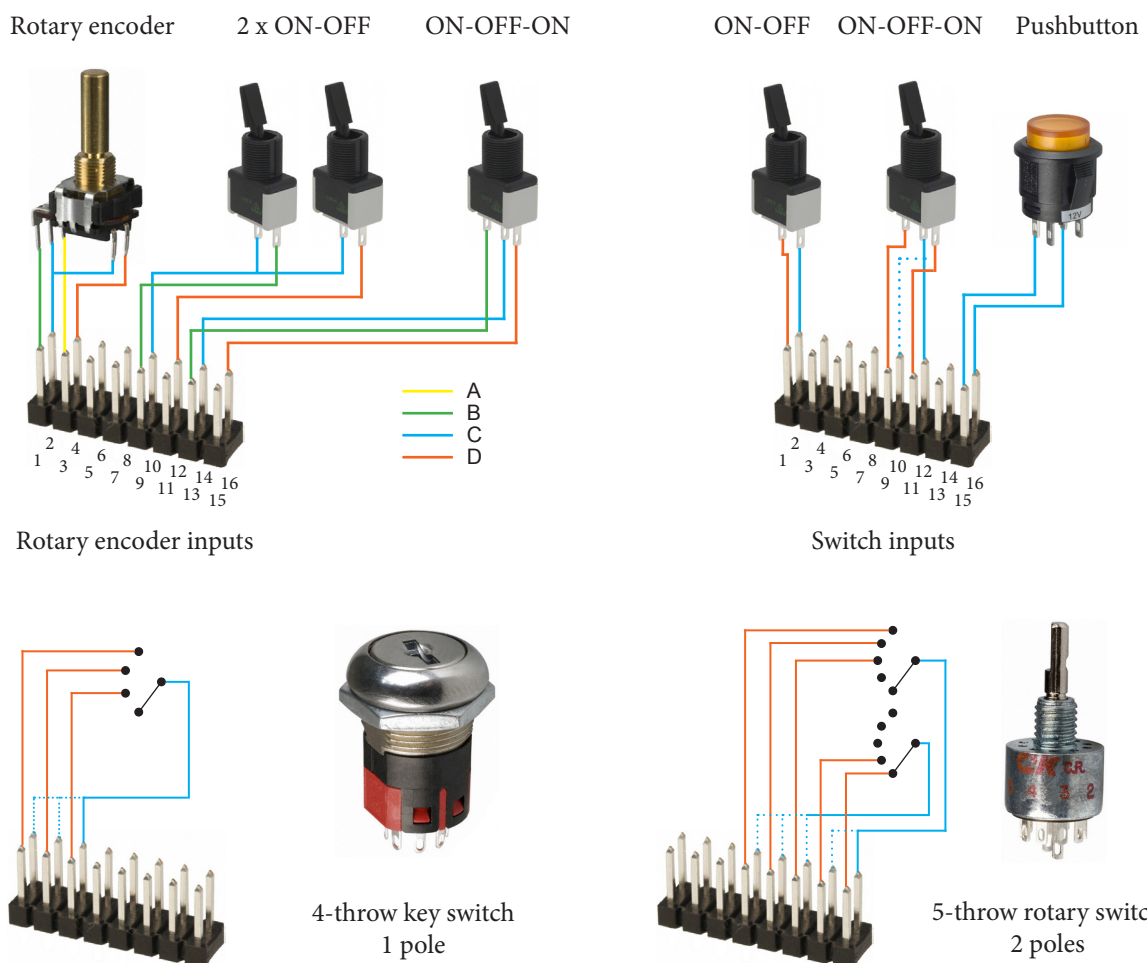
L.. - LED outputs

13. Pin Descriptions

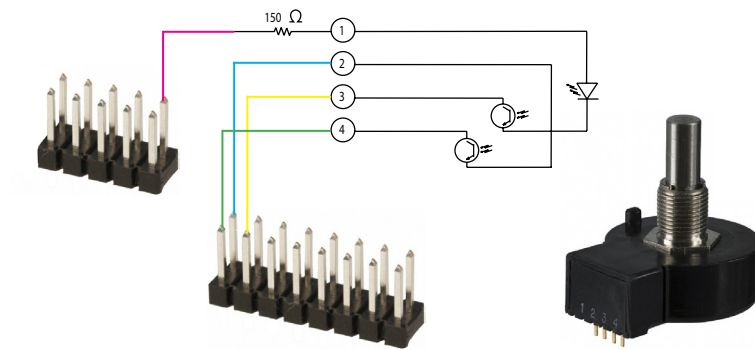
P1	P2	P3	P4	P5	P6
B1 C1	B5 C5	D17 C17	D25 C25	SDA SCL	K1 A1
A1 D1	A5 D5	D18 C18	D26 C26	Gnd +5V	K2 A2
B2 C2	B6 C6	D19 C19	D27 C27	Gnd +5V	K3 A3
A2 D2	A6 D6	D20 C20	D28 C28	Gnd +5V	K4 A4
B3 C3	B7 C7	D21 C21	D29 C29	Gnd +5V	K5 A5
A3 D3	A7 D7	D22 C22	D30 C30	Gnd +5V	K6 A6
B4 C4	B8 C8	D23 C23	D31 C31	P7 Reserved	K7 A7
A4 D4	A8 D8	D24 C24	D32 C32		K8 A8

The Rotary pins do not have an electrical connection to each other and may also not be connected externally. The switch inputs have three of the C-pins common with one of the matrix rows (in the above picture outlined by bars) and can be used externally together - see wiring examples *switches* and *multi-switches*. Multi-switches with more than 3 (4) positions must be equipped with two or more levels (poles).

14. Wiring Examples

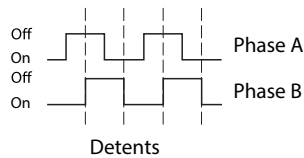


Optical Rotary Encoder

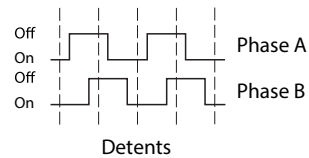


15. Hints

Mechanical encoders with detent function are available in two variants, depending on where the snap-in positions in relation to the output signals are:

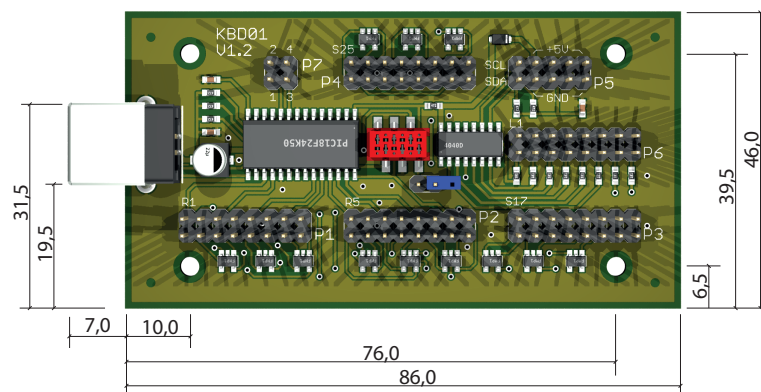


Variant 1



Variant 2

In the case of variant 1 it should be ensured that terminal A (Phase A) is correctly wired to the pin designated as „A“. For variant 2, A and B can be interchanged - it only then changes the direction of rotation.



Dimensions [mm]