Pi1541 – Rotary Encoder Board Rev. 0

Prototype Testing

# Test Setup

The Board was connected to the existing Pi1541 setup, replacing the switch board. The software running on the Pi1541 is v1.19 (not modified).

A 4-channel oscilloscope was connected to the two rotary encoder outputs and the outputs for the up and down switch. The latter outputs are active low, that means a pulse is present while low level.



Figure 1: Test setup

# Test Results

| **Test** | **Result** | **Testing** |
| --- | --- | --- |
| Turning the rotary encode CW | Proper function as a down switch (SW3) | Ok |
| Turning the rotary encoder CCW | Proper function as an up switch (SW2 | Ok |
| Pushing the rotary encode | Proper function as the select switch (SW1) | Ok |
| Pushing Switch SW4 | Proper function as a back switch | Ok |
| Pushing Switch SW5 | Proper function as an insert disk switch | Ok |
| Turning the rotary encoder fast (in every direction) | Some pulses get lost. It seems that the Pi1541 is not fast enough to follow. The more to scroll on the display, the more pulses get lost. | It does not influence the usability in a bad way.  Ok. |
| Optimizing the mono-flop timing: R3 and R4 replaced with 47k | Pulses too short for the Pi1541 to detect | Not ok. |
| Optimizing the mono-flop timing: R3 and R4 replaced with 100k | Pulses are detected by the Pi1541 software. Pulse length 33ms | Ok. |
| Optimizing the mono-flop timing: R3 and R4 replaced with 75k | Pulses are detected by the Pi1541 software. Pulse length 26ms | Ok. |

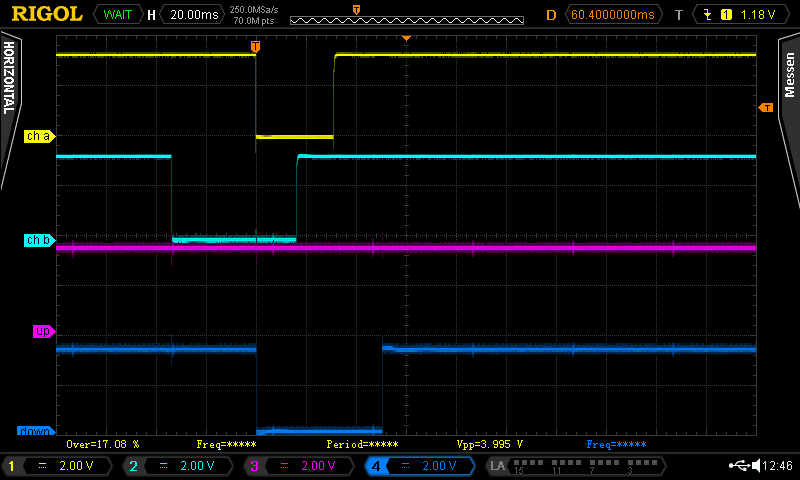


Figure 2: Turning clockwise (R3/4=150k) ⇒ Pulses on the „down-channel” (SW3)

Figure 2 shows the signals of the rotary encode (ch a and ch b) and the output signals issued in the up-switch (SW2) and down-switch channel (SW3). When turning clockwise, the (low) pulse is issued on the “down” channel.

Turning counter-clockwise is shown in Figure 3. The pulse width od the pulses is determined by the capacitors C3 and C4 (both 1u) and the resistors R3 and R4. Turning the rotary encoder too fast is resulting in “wide pulses”, the mono-flop is retriggered while the set time is elapsing, thus a pulse is lost. This shown in Figure 4. The first four ticks on the rotary encoder result in one pulse.

To increase the possible speed of the rotary encode, the resistors R3/R4 are modified. A value of 47k instead of the original 150k for these resistors results in the rotary encoder is not recognized by the Pi1541 software at all. In this case the pulse width is approximately 17ms. The resistor values were increased to 100k, the rotary encoder was effective again. The capacitors have a tolerance of 10%. Proving, that the value of 100k is suitable, two 75k resistors were soldered in. The output signals were recognized by the Pi1541 software.

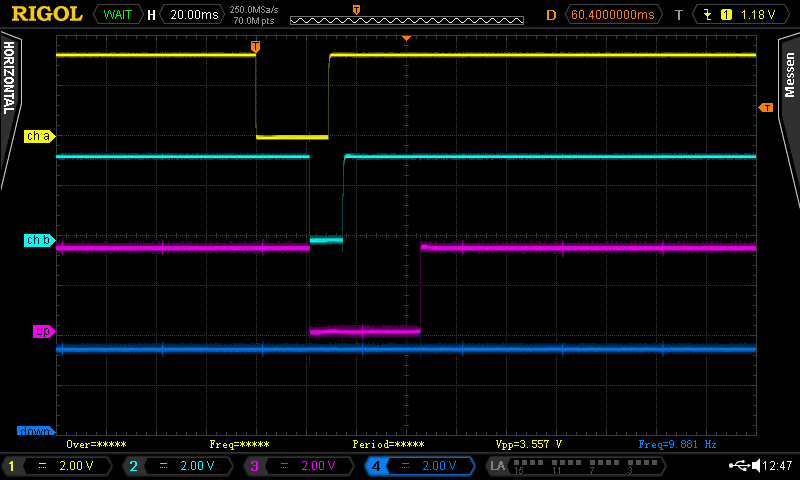


Figure 3: Turning counter-clockwise (R3/4=150k) ⇒ Pulses on the „up-channel” (SW2)

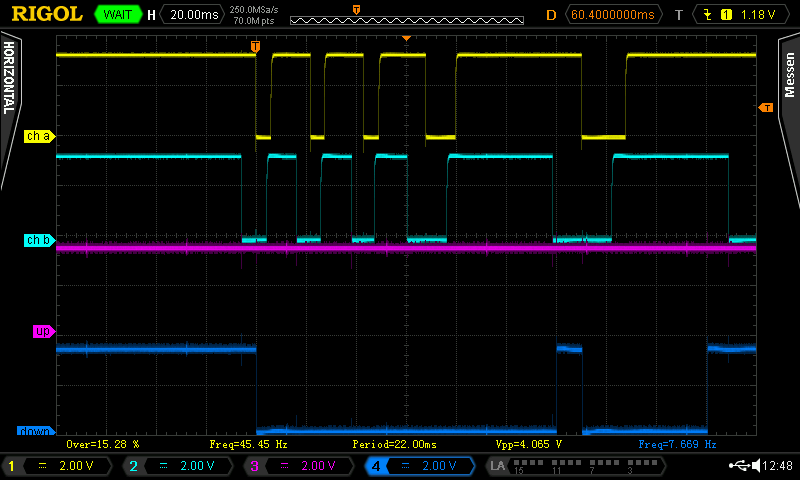


Figure 4: Turning CW too fast

# Assembly

The latches at the side of the rotary encoder were a bit tight fit, the component could be installed, though. In future revisions, the size of the holes should be increased from 2.4mm to 2.6mm.

IC3 does not have a Pin 1 mark on the top silk layer, it is only included in the documentation print out (layer 51 tDocu). This can be improved in the library.

# Conclusion

**The rotary encoder board is fully functional** and the lost pulses, when turning too fast are not considered as a problem. A rate of approximately 15 clicks per seconds can be achieved by the hardware. One complete turn of the rotary encoder are 20 clicks. In case the file/directory names are long, the scrolling is slower than if the names to be scrolled are short. The rotary encoder speeds up the navigation a lot compared to navigating with the buttons. It is much more comfortable to use.