This ULU

The *ULU.13 1x16 (de)multiplexer* can be used to address 16 memory positions. In combination with *ULU.14 Patch Panel*, this ULU can be used to make an instruction decoder. Finally, it can be used to multiplex up to 16 signals into one signal.

Used parts

Only standard parts are used:

1x casing 50*80*20mm;
2x 2mm signal connector;
2x black O-ring 9x5x2mm;
5x 4-bit data connector;
5x colored O-ring 8x5x1.5mm
1x power connector;
5x 5mm rectangular LED;
3x resistor to dim these LEDs;
4x 3x5mm bolt;
2x 5mm offset;
8x mini (SRD-5VDC-SL-C) relay;
8x Diode 1N5817.

Construction

The standard ULU specifications are applicable as specified in the datasheet *ULU.00 – Common specifications*. The build starts by taping the 8 relays together to form a solid block. The tape is only applied to the bottom and long sides. It may be necessary to sand both long sides along the pins, in order to make the relay more square instead of trapezoid.

First the ground of the relays is connected (Figure 2), using bare wire and diodes. Then the four inputs are connected. After that is done, the short connections are soldered. For that purpose, the cutted end of a LED pin can be used.

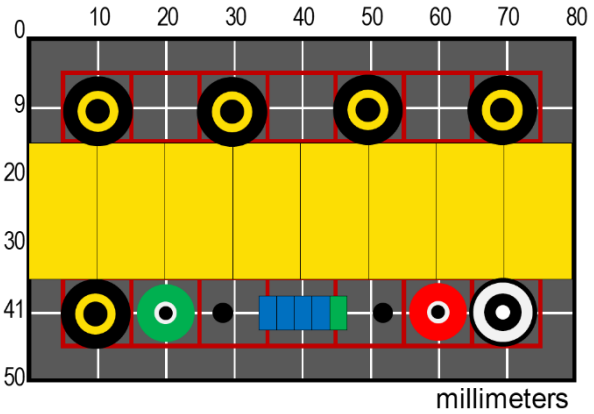

Figure 1 – Drill guide

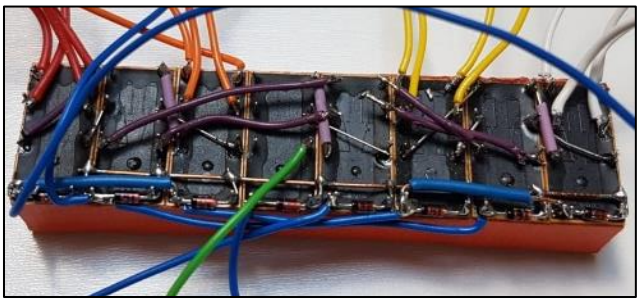

Figure 2 – Soldered relays

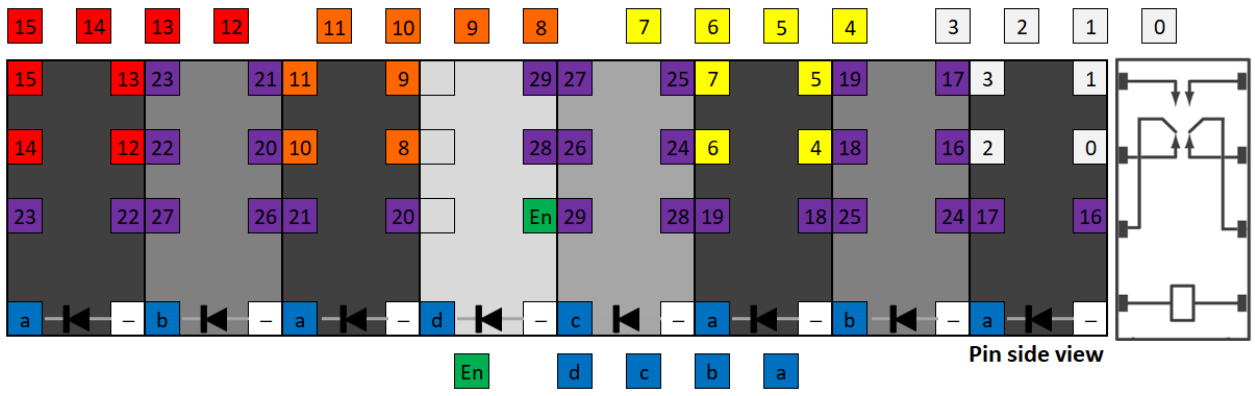

Figure 3 – Solder guide: connect the corresponding numbers

If necessary, this pin is insulated with a small piece of cable insulation. Since there is insufficient please to fit a rocker switch, the red socket is used as a power (+5V) output and connected to the green enable socket for enabling the multiplexer.

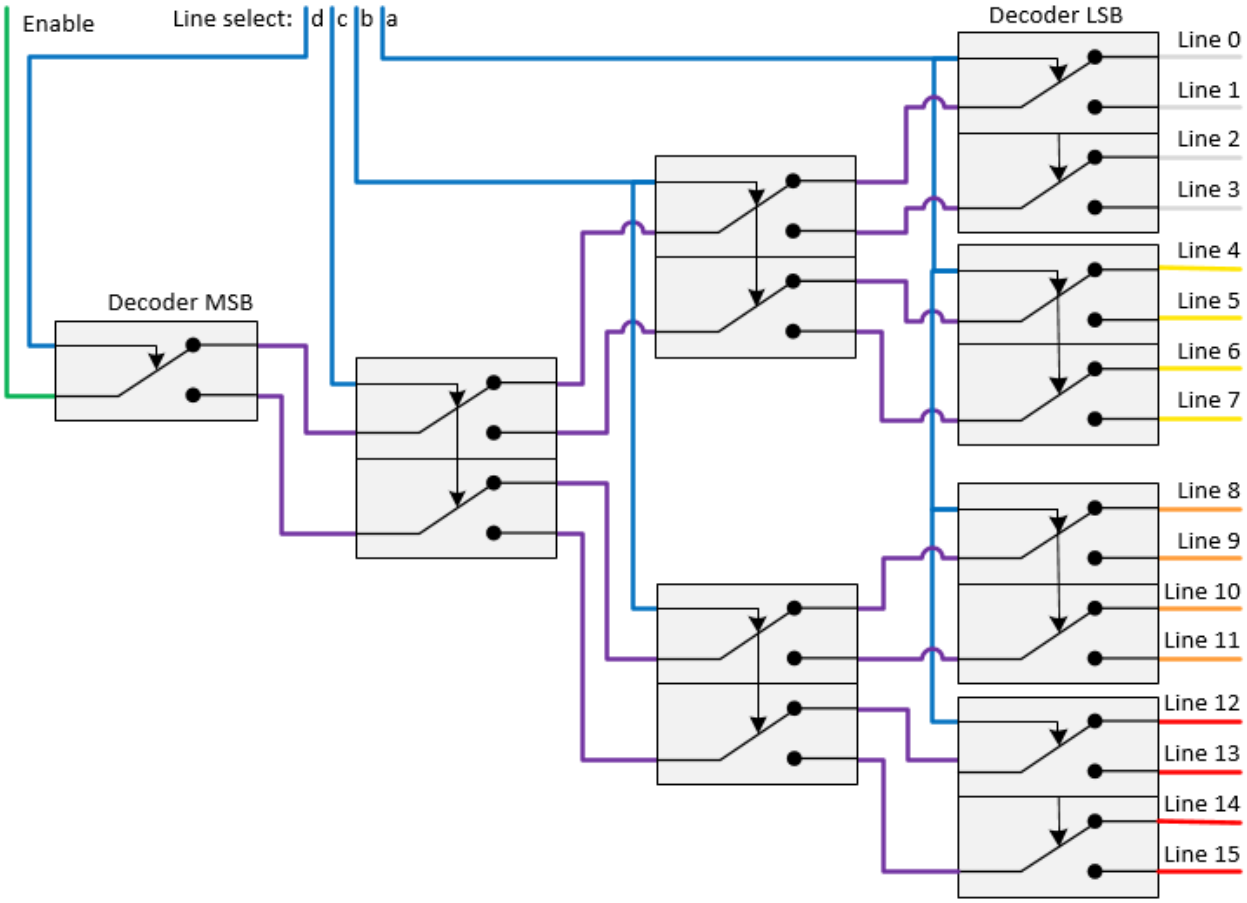

Figure 4 – Multiplexer schematic

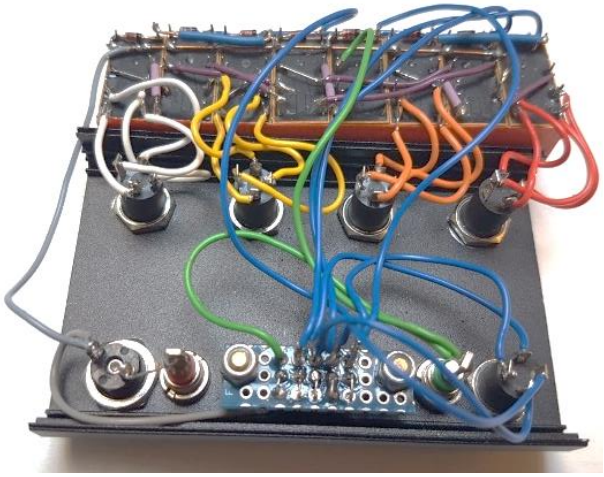

Figure 5 – ULU inside

Figure 6 – Finished ULU

Usage

A sparse matrix is applicable for an instruction decoder of a computer. Using a memory or Arduino, is not as easy as it seems. It requires a relay for every output and filling (programming) the matrix is not trivial. So, a multiplexer in combination with one or more patch panels is the best way to implement an instruction decoder.

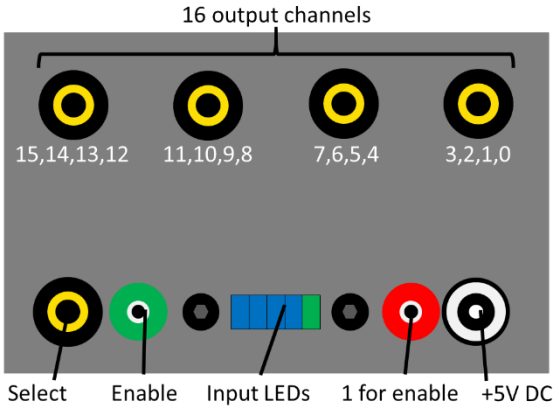

Figure 7 – Controls and connectors