Button Matrix - Documentation

Revision 1

1 Circuit Board

1.1 Schematic

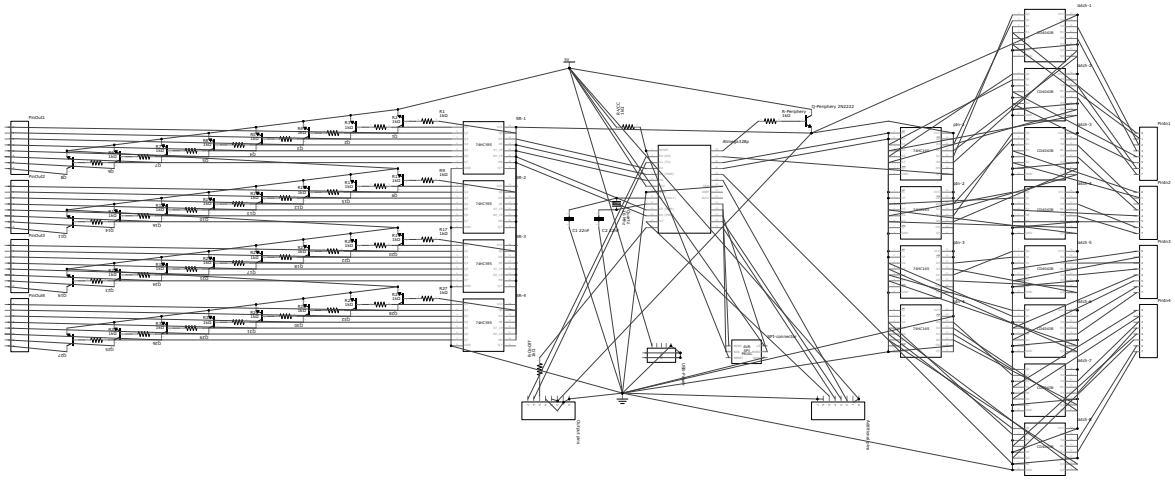

Figure 1: Schematic view of the circuit board design.

1.2 Layout - Top

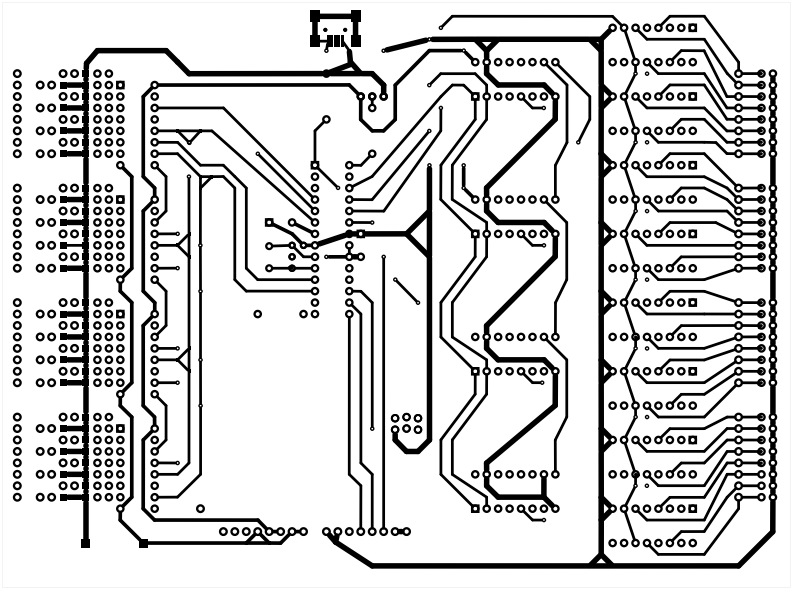

Figure 2: View of the circuit board top as seen from the top.

1.3 Layout - Bottom

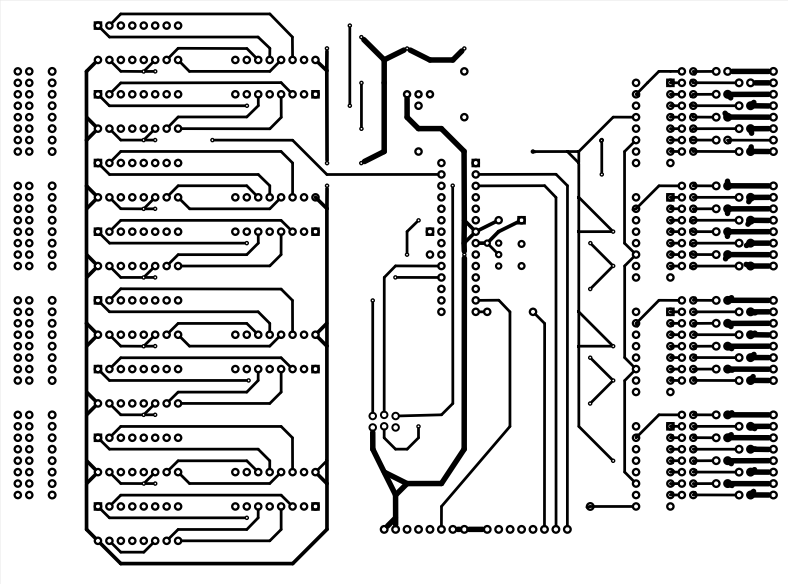

Figure 3: View of the circuit board bottom as seen from the bottom.

1.4 Layout - Hardware

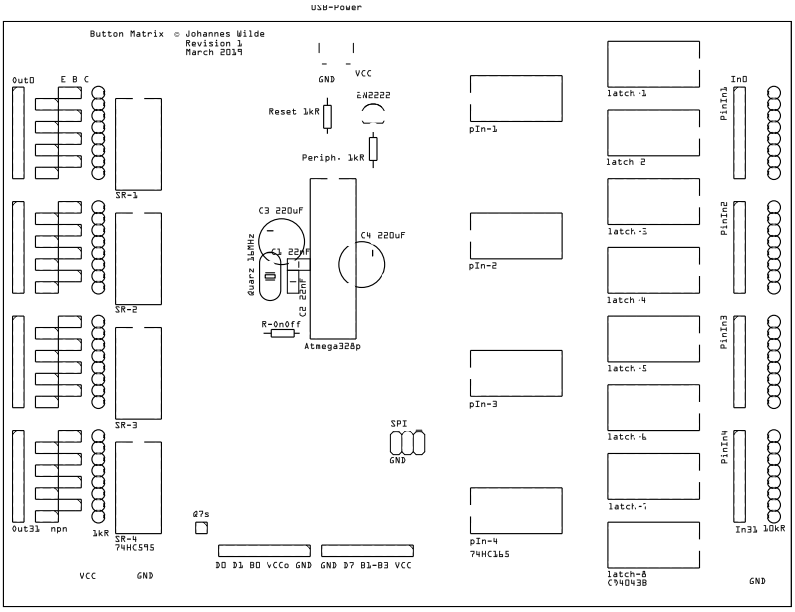

Figure 4: Positioning of the hardware as seen from the top.

1.5 Pin Connections

Table 1: Pins of the ATmega328P and their respective connections.

ATmega328P pin	connected to
VCC	VCC
GND	GND
AREF	VCC
AVCC	VCC
B0	connectorLeft - 5
B1	connectorRight - 4
B2	connectorRight - 3
B3	SPI - MOSI
B4	SPI - MISO
B5	SPI - CLK
B6	XTAL
B7	XTAL
C0	connectorRight - 2
C1	CD4043B - R_i
C2	74HC165 - CP
C3	74HC165 - $\overline{PL}$
C4	74HC165 - d_7
C5	VCCo enable
C6	$\overline{Reset}$
D0	connectorLeft - 7
D1	connectorLeft - 6
D2	74HC595 - SER
D3	74HC595 - $\overline{OE}$
D4	74HC595 - RCLK
D5	74HC595 - SRCLK
D6	74HC595 - $\overline{SRCLR}$
D7	connectorRight - 5

Table 2: Pins of the 74HC595 and their respective connections.

74HC595 pin	connected to
VCC	VCCo
GND	GND
SER	ATmega328P - D2 / 74HC595″ - q_7
$\overline{OE}$	ATmega328P - D3
RCLK	ATmega328P - D4
SRCLK	ATmega328P - D5
$\overline{SRCLR}$	ATmega328P - D6
q_7	74HC595′ - SER / NC
Q_i	Out_j-transistor base

Table 3: Pins of the 74HC165 and their respective connections.

74HC165 pin	connected to
VCC	VCCo
GND	GND
$\overline{\text{PL}}$	ATmega328P - C3
CP	ATmega328P - C2
d_7	ATmega328P - C4 / 74HC165″ - SER
$\overline{d_7}$	NC
SER	74HC165′ - d_7 / GND
$\overline{\text{CE}}$	GND
D_i	CD4043B - Q_i

Table 4: Pins of the CD4043B and their respective connections.

CD4043B pin	connected to
VCC	VCCo
GND	GND
Q_i	74HC165 - D_j
S_i	In_k
R_i	ATmega328P - C1
OE	VCCo

Table 5: Pins of the connectorLeft and their respective connections.

connectorLeft pin	connected to
0	GND
1	GND
2	VCCo
3	VCCo
4	GND
5	ATmega328P-B0
6	ATmega328P-D1
7	ATmega328P-D0

Table 6: Pins of the connectorRight and their respective connections.

connectorRight pin	connected to
0	VCC
1	VCC
2	ATmega328P-C0
3	ATmega328P-B2
4	ATmega328P-B1
5	ATmega328P-D7
6	GND
7	GND

Above notations are held rather general - for the actual connections please refer to the schematic or board layouts above.

For better understandability however some general remarks:

1. The devices containing shiftregisters [74HC595, 74HC165] are concatenated with those of the same type, i.e. one device's output is connected to the next device's input [e.g. for 74HC595 it is q_7 of the next one connected to SER of the previous one]. This is supposed to be inferred when using $'$ in above tables.
2. The last device in those concatenations is somewhat special, insofar as that for 74HC595 q_7 is NC and for 74HC165 SER is connected to GND.
   And for the first device 74HC595 SER is connected to ATmega328P - D2 and for 74HC165 d_7 is connected to ATmega328P - C4.
3. The connection of the 74HC595 to the Out is rather straightforward: the first output of the first shiftregister in the concatenation is connected to the first Out and the last output of the last shiftregister in the concatenation is connected to the last Out.
4. For In this is somewhat more complicated for **Revision 1**:

74HC165 pin	In pin
0	3
1	2
2	1
3	0
4	4
5	5
6	6
7	7

2 Hardware Description

2.1 ATmega328P

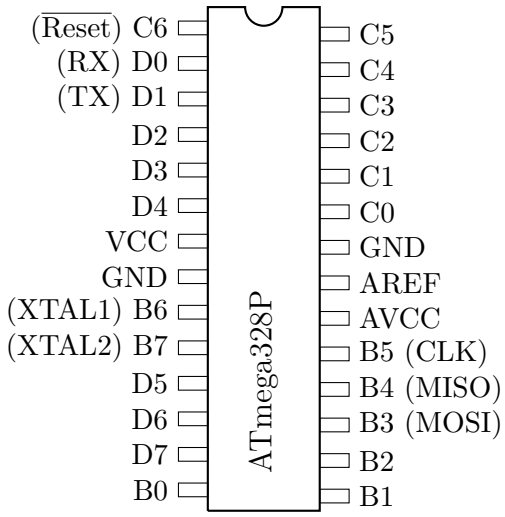

Figure 5: Schematic of the ATmega328P.

The bracketed names specify functionality assigned to the respective pins by the underlying hardware design. If one would want, these can be reconfigured for different use cases however.

2.2 74HC595 - 8-bit Shift Register

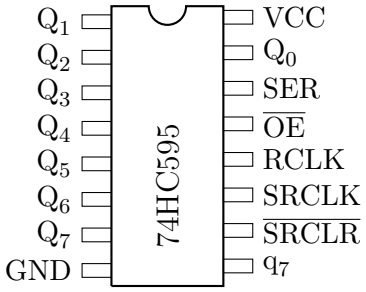

Figure 6: Schematic of the 8-bit shift register 74HC595.

Working principle:
The shiftregister has an internal [q_i, $i \in [0,7]$] and an external [Q_i, $i \in [0,7]$] 8-bit register.
On a rising edge on SRCLK the values q_i are shifted to q_{i+1} [$i \in [0,6]$, higher i first] and SER to q_0.
On a rising edge on RCLK the values of q_i are copied to Q_i [$i \in [0,7]$]. These will only be visible externally if $\overline{\text{OE}}$ is LOW [otherwise the outputs will be in a high impedance state].
A LOW on $\overline{\text{SRCLR}}$ will set q_i [$i \in [0,7]$] LOW.
VCC and GND are required for the shift-register to work.
q_7 can be used to pass the out-shifted bits on to another shift-register, if q_7 is connected to SER of the next shift-register [which will have to be clocked accordingly].

Q_i - Output i; GND - Ground; q_7 - Serial output; $\overline{\text{SRCLR}}$ - Clear internal shift-register on LOW; RCLK - Copy internal to external shift-register; SRCLK - shift in SER on rising edge; $\overline{\text{OE}}$ - Output Enable; SER - Serial input; VCC - Supply Voltage.

2.3 74HC165 - 8-bit parrallel in, serial out register

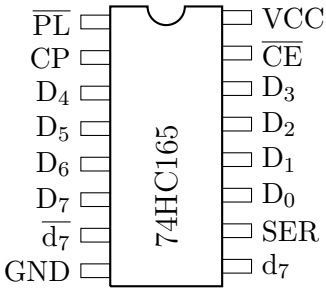

Figure 7: Schematic of the parallel in, serial out IC 74HC165.

Working principle:
The parallel load, serial out IC has an internal [d_i, $i \in [0,7]$] and an external [D_i, $i \in [0,7]$] 8-bit register.
When $\overline{\text{PL}}$ is LOW the D_i are copied to the d_i [$i \in [0,7]$] asynchronously, i.e. without the need for a clock.
When $\overline{\text{PL}}$ is HIGH the 74HC165 will function as a shift-register: on a positive edge on CP [if $\overline{\text{CE}}$ is low] the values d_i are shifted to d_{i+1} [$i \in [0,6]$, higher i first] and SER to d_0. Additionally the complementary signal of the new d_7 will be visible on $\overline{d_7}$.
VCC and GND are required for the 74HC165 to work.
d_7 can be used to pass the out-shifted bits on to another 74HC165, if d_7 is connected to SER of the next 74HC165 [which will need to have $\overline{\text{PL}}$ low, $\overline{\text{CE}}$ low and be clocked accordingly].
$\overline{\text{PL}}$ - Not Parallel Load; CP - Clock; D_i - Parallel Data In; d_7 - Serial Output; $\overline{d_7}$ - Not Serial Output; GND - Ground; SER - Serial In; $\overline{\text{CE}}$ - Not Clock Enable; VCC - Supply Voltage.

2.4 CD4043B - CMOS Quad 3-State R/S-Latches [NOR]

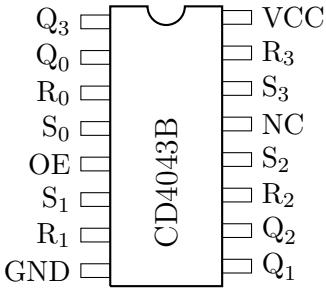

Figure 8: Schematic of the quad 3-state R/S Latch CD4043B.

Table 7: Functional states of the quad 3-state R/S Latch CD4043B [$i \in [0, 3]$].

S_i	R_i	OE	Q_i
X	X	LOW	high impedance
LOW	LOW	HIGH	unchanged
HIGH	LOW	HIGH	HIGH
LOW	HIGH	HIGH	LOW
HIGH	HIGH	HIGH	HIGH

Working principle:
OE controls whether the ouputs Q_i [$i \in [0, 3]$] are connected [OE is HIGH] or in a high-impedance state [like an open circuit; OE is LOW].
Each separate output Q_i [$i \in [0, 3]$] can be set to HIGH [S_i HIGH] or reset to LOW [R HIGH] seperately and asynchronously.
In case S_i and R_i are HIGH simultaneously, the respective output Q_i will read HIGH.
NC is not connected internally.

Q_i - Output i; R_i - Reset i; S_i - Set i; OE - Output Enable; GND - Ground; NC - Not Connected; VCC - Supply Voltage.

2.5 LED

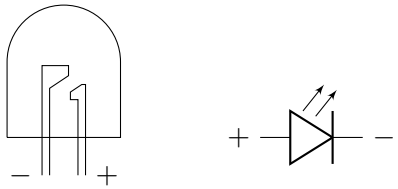

Figure 9: Schematic view of a LED and its representation in circuit diagrams.

2.6 npn-Transistor

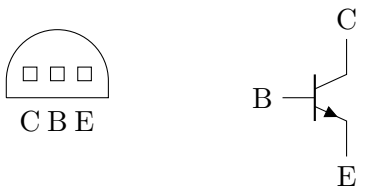

Figure 10: Schematic top view onto a npn-transistor and its representation in circuit diagrams.

2.7 Periphery

3 SPI-Programming

The ATmega328P can be programmed using the serial peripheral interface [SPI].

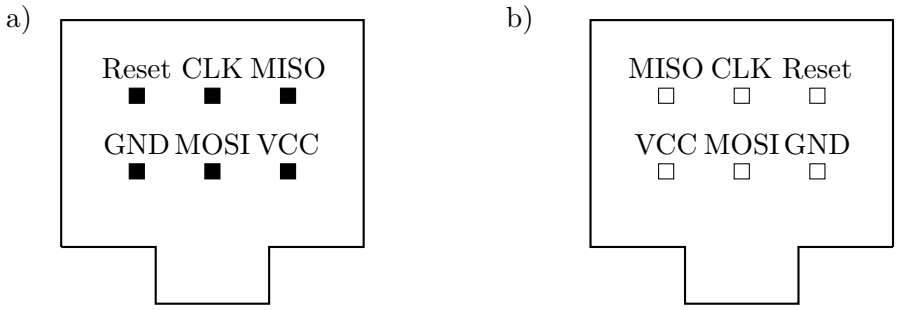

Figure 11: SPI connectors - a) connector, b) cable.

I used an Arduino Uno programmed with the "Arduino ISP" sketch [under Examples in the Arduino IDE] to program my separate ATmega328P.

In order for that to work one will have to keep the Reset pin of the programming Arduino Uno HIGH with a capacitor and connect the Reset pin of the device to be programmed with the pin as specified in the "Arduino ISP" sketch via **#define** RESET 10 [this being pin D10 per default].

Other than that each pin of the programmer will have to be connected with the pin of the same name of the programmee [i.e. e.g. MOSI-programmer → MOSI-programmee].