

TL72_LCD_8Bit_UART_ADC

Dean Schneider

31.05.2025



Abbildung 1: Aufbau auf Breadboard

Die Betriebsspannung beträgt $U = 5\text{ V}$.

Vorgegebene Spannungen die mit Widerstandskombinationen erstellt werden sollen:

$$U_{A0} \approx 0,8\text{ V}$$

$$U_{A1} \approx 1,6\text{ V}$$

$$U_{A2} \approx 2,3\text{ V}$$

$$U_{A3} \approx 3,7\text{ V}$$

$$U_{A4} \approx 4,3\text{ V}$$

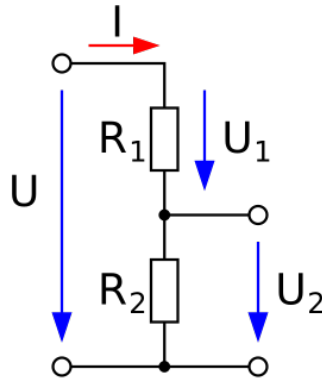


Abbildung 2: Spannungsteiler (Quelle: wikipedia.org/wiki/Spannungsteiler)

Diese Formel berechnet die Spannung an R_2 (vgl. Abbildung 2), welche vom ATmega328P gelesen wird.

$$U_2 = U \cdot \frac{R_2}{R_1 + R_2}$$

Zur Verfügung stehen diese Widerstände:

$$R \in \{100 \, \Omega; 220 \, \Omega; 330 \, \Omega; 1 \, \text{k}\Omega; 5,1 \, \text{k}\Omega\}$$

Die Toleranzen werden hier vernachlässigt, da die Widerstände eine geringe Toleranz von 1 % haben und die Aufgabe nur ungefähre Spannungen vorgibt.

$U_{A0} = 5 \, \text{V} \cdot \frac{1 \, \text{k}\Omega}{5,1 \, \text{k}\Omega + 1 \, \text{k}\Omega} \approx 0,82 \, \text{V}$	für $R_1 = 5,1 \, \text{k}\Omega; R_2 = 1 \, \text{k}\Omega$
$U_{A1} = 5 \, \text{V} \cdot \frac{100 \, \Omega}{220 \, \Omega + 100 \, \Omega} \approx 1,563 \, \text{V}$	für $R_1 = 220 \, \Omega; R_2 = 100 \, \Omega$
$U_{A2} = 5 \, \text{V} \cdot \frac{1 \, \text{k}\Omega}{1 \, \text{k}\Omega + 1 \, \text{k}\Omega} = 2,5 \, \text{V}$	für $R_1 = 1 \, \text{k}\Omega; R_2 = 1 \, \text{k}\Omega$
$U_{A3} = 5 \, \text{V} \cdot \frac{330 \, \Omega}{100 \, \Omega + 330 \, \Omega} \approx 3,837 \, \text{V}$	für $R_1 = 100 \, \Omega; R_2 = 330 \, \Omega$
$U_{A4} = 5 \, \text{V} \cdot \frac{5,1 \, \text{k}\Omega}{1 \, \text{k}\Omega + 5,1 \, \text{k}\Omega} \approx 4,18 \, \text{V}$	für $R_1 = 1 \, \text{k}\Omega; R_2 = 5,1 \, \text{k}\Omega$

Um aus dem 10 Bit Messwert x eine Spannung zu berechnen wird diese Formel verwendet:

$$U_A = U \cdot \frac{x}{1023} \quad x \in [0; 1023]$$

```

1  /* TI_71_LCD_8Bit_UART
2  * LCD      Adruino      MC
3  * RS       13           PB5
4  * R/!W     12           PB4
5  * E        11           PB3
6  * D0       10           PB2
7  * D1       9            PB1
8  * D2       8            PB0
9  * D3       7            PD7
10 * D4       6            PD6
11 * D5       5            PD5
12 * D6       4            PD4
13 * D7       3            PD3
14 *
15 * Spannungsteiler
16 *          A0           PC0
17 *          A1           PC1
18 *          A2           PC2
19 *          A3           PC3
20 *          A4           PC4
21 */
22
23 #define info "TI_71_LCD_8Bit_UART"
24 #include <LiquidCrystal.h>
25 /* Create object named lcd of the class LiquidCrystal */
26 LiquidCrystal lcd(13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3); /* For 8-bit mode */
27 //LiquidCrystal lcd(13, 12, 11, 6, 5, 4, 3); /* For 4-bit mode */
28
29 unsigned char Character1[8] = { 0x04, 0x1F, 0x11, 0x11, 0x1F, 0x1F, 0x1F, 0x1F };
30 /* Custom Character 1 */
31 unsigned char Character2[8] = { 0x01, 0x03, 0x07, 0x1F, 0x1F, 0x07, 0x03, 0x01 };
32 /* Custom Character 2 */
33
34 void setup() {
35     lcd.begin(16, 2); /* Initialize 16x2 LCD */
36     lcd.clear(); /* Clear the LCD */
37     lcd.createChar(0, Character1); /* Generate custom character */
38     lcd.createChar(1, Character2);
39     lcd.setCursor(0, 0); /* Set cursor to column 0 row 0 */
40     lcd.print(info); /* Print data on display */
41
42     Serial.begin(115200); // highspeed UART bertragung per USB
43     while (!Serial); // Warte bis Serielle Schnittstelle aktiv ist
44     delay(2000);
45     Serial.println(info);
46     Serial.println("Start in 3 sek!"); /* Print data on display */
47     lcd.setCursor(0, 1); /* Set cursor to column 0 row 0 */
48     lcd.print("Start in 3 sek!"); /* Print data on display */
49     delay(3000);
50
51     lcd.clear();
52     lcd.setCursor(0, 0);
53     lcd.print("Bitte Analog-Pin");
54     lcd.setCursor(0, 1);
55     lcd.print("via UART senden");
56 }
57
58 // Liest die 10 fache Spannung eines Analogen eingangs aus. (adc_index=0
59 // entspricht A0)
60 uint8_t read_voltage_10x(uint8_t adc_index) {
61     // A0 = 14
62     const uint8_t adc_port = adc_index + 14;

```

```

60
61 // 50V anstatt von 5V da es sich um den 10 fachen Wert handelt
62 return (50 * (uint16_t) analogRead(adc_port)) / 1023;
63 }
64
65 void loop() {
66     if (Serial.available()) {
67         lcd.clear();
68         lcd.setCursor(0, 0);
69
70         // '0' - '4'
71         const char adc_selection = Serial.read();
72
73         if (adc_selection < '0' || adc_selection > '4') {
74             Serial.println("Bitte einen Wert zwischen 0 und 4 eingeben.");
75             return;
76         }
77
78         // ASCII: '0' = 48
79         // char zu int umwandeln
80         const uint8_t adc_selection_uint = adc_selection - '0';
81
82         Serial.print("Spannung an A");
83         Serial.println(adc_selection_uint);
84
85         lcd.print("Spannung an A");
86         lcd.print(adc_selection_uint);
87
88         const uint8_t voltage_10x = read_voltage_10x(adc_selection_uint);
89
90         // Erste Zahl vor dem Komma
91         const uint8_t voltage_first_digit = voltage_10x / 10;
92
93         // Erste Zahl nach dem Komma
94         const uint8_t voltage_after_comma = voltage_10x % 10;
95
96         Serial.print(voltage_first_digit);
97         Serial.print(',');
98         Serial.print(voltage_after_comma);
99         Serial.println('V');
100
101         lcd.setCursor(0, 1);
102         lcd.print(voltage_first_digit);
103         lcd.print(',');
104         lcd.print(voltage_after_comma);
105         lcd.print('V');
106     }
107 }

```