

Arduino LCD Shield Demonstrationen

1	Vorbemerkungen.....	2
1.1	Beschreibung.....	2
1.2	Bemerkungen.....	2
2	Demonstration LCD_01.....	3
2.1	Zielstellung.....	3
2.2	Lösung.....	3
3	Demonstration LCD_02.....	4
3.1	Zielstellung.....	4
3.2	Lösung.....	5
4	Demonstration LCD_03.....	6
4.1	Zielstellung.....	6
4.2	Lösung.....	6

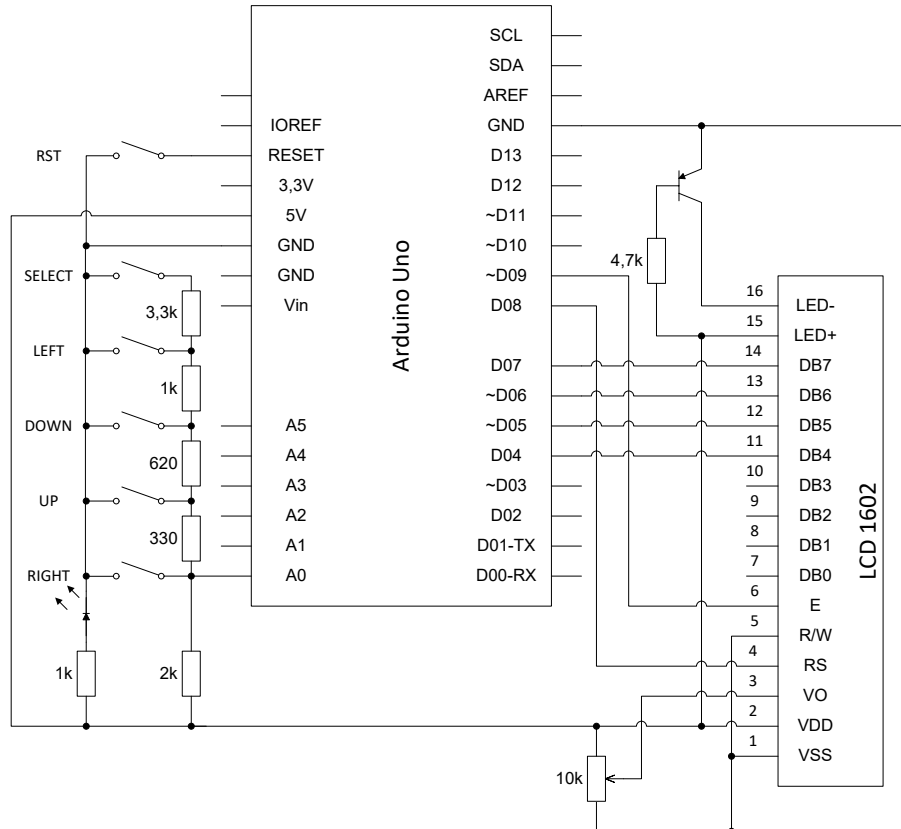
1 Vorbemerkungen

[\(Inhalt\)](#)

1.1 Beschreibung

[\(Thema\)](#)

Das Arduino LCD Shield (VMA203) für den Arduino Uno kombiniert eine 2-Zeilen LCD-Anzeige mit einem 6-Tasten Keypad.



1.2 Bemerkungen

[\(Thema\)](#)

Die Taste RESET auf dem Keypad übernimmt die Funktion der RESET Taste vom Arduino, da sie durch das Shield nicht mehr zugänglich ist.

Das Keypad belegt dabei nur einen analogen Pin des Arduino. Für die LCD-Anzeige sind 6 Pins erforderlich.

Es bleiben noch folgende Pins für eigenen Projekte frei:

00, 01, 02, 03, 10, 11, 12, 13, A1, A2, A3, A4, A5.

Die Programmierung basiert auf der Bibliothek „LiquidCrystal.h“. Die Demonstrationen LCD_2 und LCD_3 sind an das Programm des Manuels von Velleman angelehnt.

2 Demonstration LCD_01

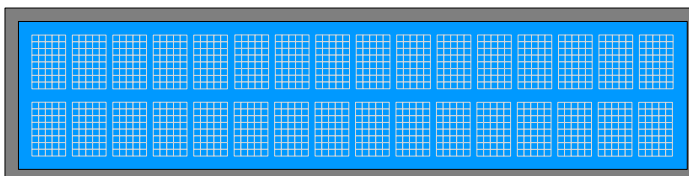
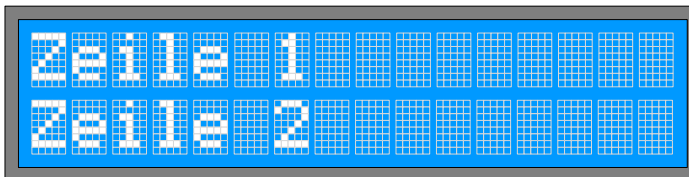
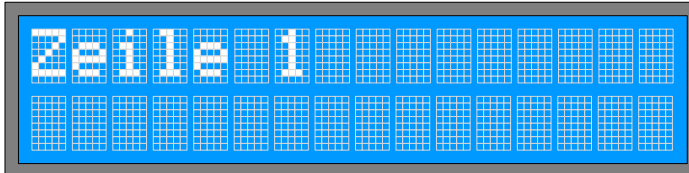
[\(Inhalt\)](#)

2.1 Zielstellung

[\(Thema\)](#)

In der ersten Demonstration geht es zunächst nur um die LCD-Anzeige.

Im Abstand von einer Sekunde sind wiederholt nacheinander die folgenden Anzeigen zu sehen.



2.2 Lösung

[\(Thema\)](#)

```
LCD_01
1  #include <LiquidCrystal.h>
2
3  const int PAUSE = 1000;
4  const int RS = 8;
5  const int ENABLE = 9;
6  const int DB4 = 4;
7  const int DB5 = 5;
8  const int DB6 = 6;
9  const int DB7 = 7;
10 LiquidCrystal lcd(RS, ENABLE, DB4, DB5, DB6, DB7);
11
12 void setup() {
13     lcd.begin(16, 2);
14 }
15
16 void loop() {
17     lcd.setCursor(0, 0);
18     lcd.print("Zeile 1");
19     delay(PAUSE);
20     lcd.setCursor(0, 1);
21     lcd.print("Zeile 2");
22     delay(PAUSE);
23     lcd.clear();
24     delay(PAUSE);
25 }
```

3 Demonstration LCD_02

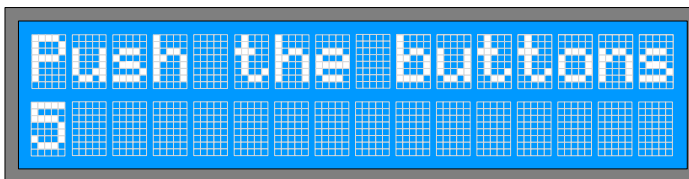
[\(Inhalt\)](#)

3.1 Zielstellung

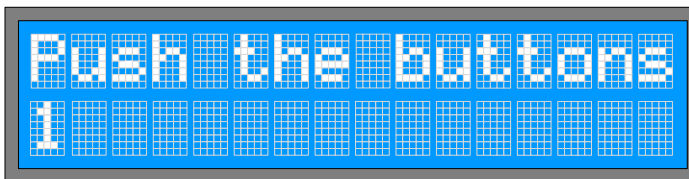
[\(Thema\)](#)

Die zweite Demonstration fragt den Zustand des Keypads ab und zeigt den jeweiligen Zustand an. Es sollen folgende Festlegungen gelten:

- 0: RIGHT,
- 1: UP,
- 2: DOWN,
- 3: LEFT,
- 4: SELECT,
- 5: Keine Taste.



Keine Taste gedrückt



Taste UP gedrückt

3.2 Lösung (Thema)

```
LCD_02
1  #include <LiquidCrystal.h>
2
3  const int PAUSE = 500;
4  const int RS = 8;
5  const int ENABLE = 9;
6  const int DB4 = 4;
7  const int DB5 = 5;
8  const int DB6 = 6;
9  const int DB7 = 7;
10
11 const int BTN_RIGHT = 0;
12 const int BTN_UP = 1;
13 const int BTN_DOWN = 2;
14 const int BTN_LEFT = 3;
15 const int BTN_SELECT = 4;
16 const int BTN_NONE = 5;
17
18 int lcdKey = 0;
19 int adcKey = 0;
20
21 LiquidCrystal lcd(RS, ENABLE, DB4, DB5, DB6, DB7);
22
23 void setup() {
24     lcd.begin(16, 2);
25     lcd.setCursor(0, 0);
26     lcd.print("Push the buttons");
27 }
28
29 void loop() {
30     lcdKey = readLcdButtons();
31     lcd.setCursor(0, 1);
32     lcd.print(String(lcdKey));
33     delay(PAUSE);
34     lcd.setCursor(0, 1);
35     lcd.print(" ");
36     delay(PAUSE / 10);
37 }
38
39 int readLcdButtons() {
40     adcKey = analogRead(A0);
41     if (adcKey > 1000)
42         return BTN_NONE;
43     if (adcKey < 50)
44         return BTN_RIGHT;
45     if (adcKey < 195)
46         return BTN_UP;
47     if (adcKey < 280)
48         return BTN_DOWN;
49     if (adcKey < 555)
50         return BTN_LEFT;
51     if (adcKey < 790)
52         return BTN_SELECT;
53     return BTN_NONE;
54 }
```

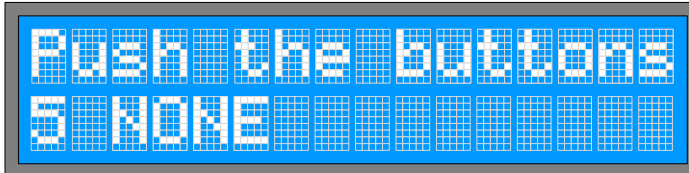
4 Demonstration LCD_03

[\(Inhalt\)](#)

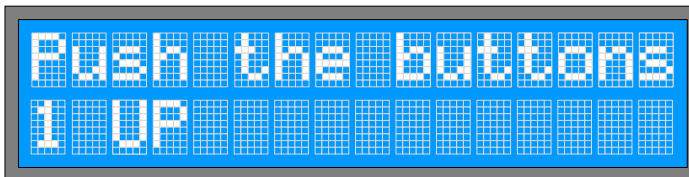
4.1 Zielstellung

[\(Thema\)](#)

Die Demonstration LCD_3 stellt eine Erweiterung von LCD_2 dar. Zusätzlich zur Ziffer für den Zustand erfolgt auch die Anzeige der Tastenbezeichnung.



Keine Taste gedrückt.



Taste Up gedrückt.

4.2 Lösung

[\(Thema\)](#)

```
LCD_03
1  #include <LiquidCrystal.h>
2
3  const int PAUSE = 500;
4  const int RS = 8;
5  const int ENABLE = 9;
6  const int DB4 = 4;
7  const int DB5 = 5;
8  const int DB6 = 6;
9  const int DB7 = 7;
10
11 const int BTN_RIGHT = 0;
12 const int BTN_UP = 1;
13 const int BTN_DOWN = 2;
14 const int BTN_LEFT = 3;
15 const int BTN_SELECT = 4;
16 const int BTN_NONE = 5;
17
18 int lcdKey = 0;
19 int adcKey = 0;
20
21 LiquidCrystal lcd(RS, ENABLE, DB4, DB5, DB6, DB7);
22
23 void setup() {
24   lcd.begin(16, 2);
25   lcd.setCursor(0, 0);
26   lcd.print("Push the buttons");
27 }
```

```

29 void loop() {
30   lcdKey = readLcdButtons();
31   showKey(lcdKey);
32 }
33
34 int readLcdButtons() {
35   adcKey = analogRead(A0);
36   if (adcKey > 1000)
37     return BTN_NONE;
38   if (adcKey < 50)
39     return BTN_RIGHT;
40   if (adcKey < 195)
41     return BTN_UP;
42   if (adcKey < 280)
43     return BTN_DOWN;
44   if (adcKey < 555)
45     return BTN_LEFT;
46   if (adcKey < 790)
47     return BTN_SELECT;
48   return BTN_NONE;
49 }

51 void showKey(int Key) {
52   lcd.setCursor(0, 1);
53   lcd.print(String(Key));
54   lcd.setCursor(2, 1);
55   switch (Key) {
56     case 0:
57       lcd.print("RIGHT");
58       break;
59     case 1:
60       lcd.print("UP");
61       break;
62     case 2:
63       lcd.print("DOWN");
64       break;
65     case 3:
66       lcd.print("LEFT");
67       break;
68     case 4:
69       lcd.print("SELECT");
70       break;
71     case 5:
72       lcd.print("NONE");
73       break;
74   }
75   delay(PAUSE);
76   lcd.setCursor(0, 1);
77   lcd.print(" ");
78   delay(PAUSE / 10);
79 }

```