

PROTOKOLL

zur Laborübung

AVR Matrixtastatur



Gruppe / Klasse 5 / 4BHELS	Protokollführer HOFSTÄTTER A.	Unterschrift
Übungs- / Abgabedatum 3. März 2015 3. März 2015	Mitarbeiter BIEHL S.	Unterschrift
Lehrer CRHA	Mitarbeiter	Unterschrift
Note	Mitarbeiter	Unterschrift

AVR Matrixtastatur

ATmega32U4

Verwendete Geräte

Nr.	Gerätebezeichnung	Hersteller	Typ	Platznummer
1.	Oszilloskop	Tektronix	TDS 1001B	-

Verwendete Programme

Nr.	Name	Version
1.	CodeBlocks	13.12
2.	DFU-Programmer	1.2.2

1 Inhaltsverzeichnis

<u>1</u>	<u>INHALTSVERZEICHNIS</u>	<u>2</u>
<u>2</u>	<u>AUFGABENSTELLUNG</u>	<u>3</u>
<u>3</u>	<u>PORT-PINBELEGUNG.....</u>	<u>4</u>
<u>3.1</u>	<u>SCHALTUNG.....</u>	<u>4</u>
<u>3.2</u>	<u>PROGRAMMLISTING</u>	<u>4</u>
<u>4</u>	<u>ABBILDUNGSVERZEICHNIS</u>	<u>6</u>
<u>5</u>	<u>PROGRAMMLISTINGVERZEICHNIS</u>	<u>6</u>

2 Aufgabenstellung

Folgende Aufgabenstellungen waren zu erledigen.

Mit Hilfe eines Mikrokontrollers war auszuwerten welche Taste einer 3x3 Matrixtastatur gedrückt wurde. Bei Druck der Taste „1“ wurde die 1. LED am PORT B eingeschaltet, bei Taste „2“ die 2. LED, usw.

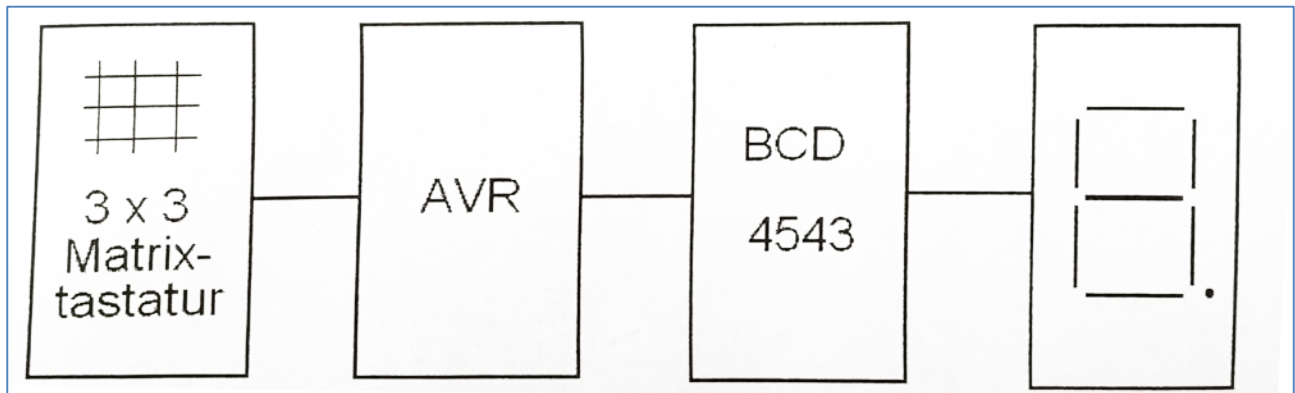


Abbildung 1. – Blockschaltbild

Statt oben gezeigter 7-Segment Anzeige mit Treiber IC wurden einzelne LEDs verwendet.

Das Grundprinzip lautet, jeder Spalte und jeder Zeile der Matrixtastatur einen PORT-Pin zuzuweisen. Dann werden nacheinander (z.B.) einzelne Zeilen (Ausgänge) auf logisch „LOW“ geschaltet und alle Spalten (Eingänge) auf logisch „LOW“ abgefragt.

Die Taste an welcher sich die gesetzte „LOW“-Zeile und die ermittelte „LOW“-Spalte schneiden wurde gedrückt.

3 Port-Pinbelegung

Die einzelnen Zeilen und Spalten wurden wie folgt den einzelnen Ports zugewiesen.

PD5	PD3	PD4	
1	2	3	-- PD1 (Ausgang)
4	5	6	-- PD2 (Ausgang)
7	8	9	-- PD0 (Ausgang)

Abbildung 2. – Pinbelegung an der Matrixtastatur

3.1 Schaltung

Alle 6 Pins der Matrixtastatur (3 Spalten, 3 Zeilen) wurden per mit Jumper-Kabel aufs Steckbrett geführt. Anschließend wurden die einzelnen Pins des PORTD vom Mikrokontroller über eine Erweiterungs-Peripherie Einheit mit einem Flachbandkabel ebenfalls aufs Steckbrett geführt und mit der Tastatur richtig verdrahtet.

3.2 Programmlisting

```
1 | #include <avr/io.h> // Include File für IO Definitionen
2 | #include <avr/interrupt.h> // Include File für sei(), cli()
3 | #include <util/delay.h>
4 |
5 | unsigned char matrix(void)
6 | {
7 |     unsigned char taste=0;
8 |
9 |     PORTD |= 0b00000110; //Setzt PB1 UND PB2 auf 1
10 |    PORTD &= ~0b11111110; //Setzt PB0 auf 0
11 |
12 |    _delay_ms(1); //Stellt Änderung sicher
13 |
14 |    if (!(PIND & 0b00100000)) taste=7; //PD5=0, daher Taste 7
15 |    else if (!(PIND & 0b00010000)) taste=8; //PD4=0, daher Taste 8
16 |    else if (!(PIND & 0b00001000)) taste=9; //PD3=0, daher Taste 9
17 |
18 |    PORTD |= 0b00000001; //Setzt PB0 auf 1
19 |    PORTD &= ~0b11111101; //setzt PB1 auf 0
20 |
21 |    _delay_ms(1);
22 |
23 |    if (!(PIND & 0b00100000)) taste=4; //PD5=0, daher Taste 4
24 |    else if (!(PIND & 0b00010000)) taste=5; //PD4=0, daher Taste 5
25 |    else if (!(PIND & 0b00001000)) taste=6; //PD3=0, daher Taste 6
26 |
27 |    PORTD |= 0b00000010; //Setzt PB1 auf 1
28 |    PORTD &= ~0b11111011; //Setzt PB2 auf 0
29 |
30 |    _delay_ms(1);
31 |
32 |    if (!(PIND & 0b00100000)) taste=1; //PD5=0, daher Taste 1
33 |    else if (!(PIND & 0b00010000)) taste=2; //PD4=0, daher Taste 2
34 |    else if (!(PIND & 0b00001000)) taste=3; //PD3=0, daher Taste 3
35 |
36 |    return taste;
37 | }
38 |
39 | int main(void)
40 | {
41 |     DDRD = 0b00000111; //Tastatur - PD0..PD2 - Ausgang, PD3..PD7 - Eingang
42 |     PORTD = 0xFF; //Tastatur - PD0..PD2 - High setzen, PD3..PD7 - Pull-Up
43 |
44 |     DDRB = 0xFF; //LEDs als Ausgang
45 |     PORTB = 0; //LEDs -> 0
```

```

46 |
47 | while(1)
48 | {   PORTB = 0x00;      //Am Beginn jeder Schleifenwiederholung PORTB 0 setzen.
49 |   switch(matrix())
50 |   {   case 1 : PORTB |= (1<<PB0); break;
51 |       case 2 : PORTB |= (1<<PB1); break;
52 |       case 3 : PORTB |= (1<<PB2); break;
53 |       case 4 : PORTB |= (1<<PB3); break;
54 |       case 5 : PORTB |= (1<<PB4); break;
55 |       case 6 : PORTB |= (1<<PB5); break;
56 |       case 7 : PORTB |= (1<<PB6); break;
57 |       case 8 : PORTB |= (1<<PB7); break;
58 |       case 9 : PORTB = 0xFF;      break;
59 |   }
60 | }
61 |

```

Listing 1. – C Code zum Auslesen einer Matrixtastatur

4 Abbildungsverzeichnis

Abbildung 1. – Blockschaltbild	3
Abbildung 2. – Pinbelegung der Matrixtastatur	4

5 Programmlistingverzeichnis

Listing 1. – C Code zum Auslesen einer Matrixtastatur	5
--	---