

1 Kannlisten Aufgaben

1.1 Aufgaben 0 bis 5

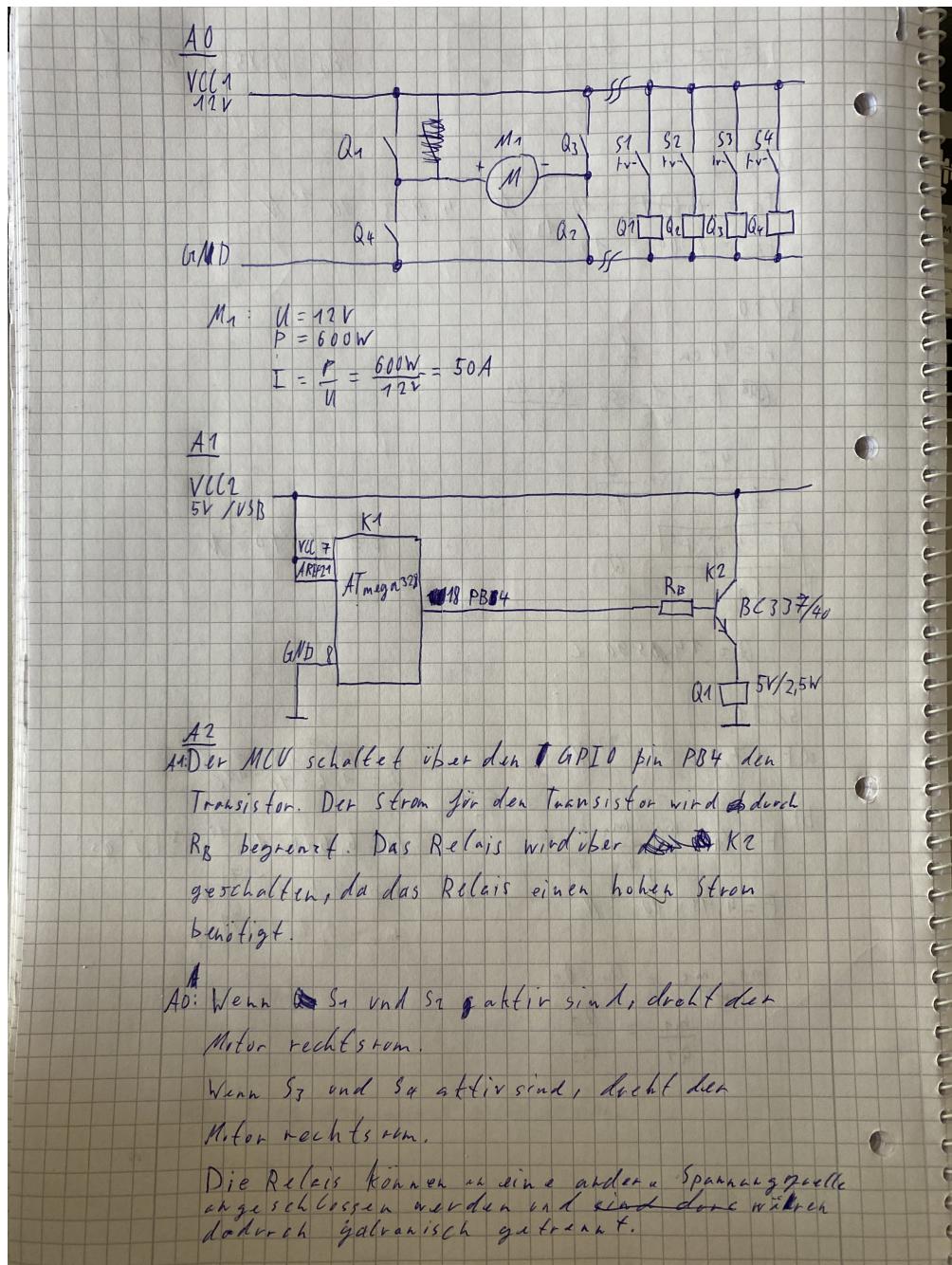


Abbildung 1: Aufgaben 0 bis 2

A3

geg.

$$U = 5V$$

$$P_{Q1} = 2,5W$$

$$U_{BE} = 0,85V$$

$$R_B = 350$$

$$U_{CEsat} = 0,7V$$

~~$$U_{Hmin} = 2,4V$$~~

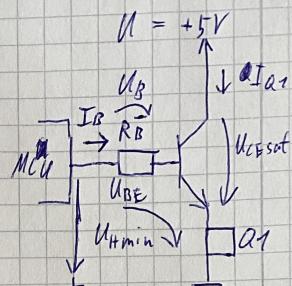
~~$$I_{Hmax} = 10mA$$~~

ges.

$$R_B$$

Anforderungen:

$$I_{B1} < I_{Hmax}$$



~~$$I_{Q1} = P_{Q1} / U = 2,5W / 5V = 0,5A$$~~

~~$$I_{Q1} = P_{Q1} / U_{CE} = 2,5W / 0,7V = 3,57A$$~~

$$R_{Q1} = \frac{U_{Q1}^2}{P_{Q1}} = \frac{(5V)^2}{2,5W} = 10\Omega$$

für $U_{Q1} = U - U_{CEsat}$:

$$I_{Q1} = \frac{U_{Q1}}{R_{Q1}} = \frac{5V - 0,7V}{10\Omega} = 430mA$$

$$\begin{aligned} P_{Q1} &= U_{Q1} \cdot I_{Q1} \\ \Rightarrow P_{Q1} &= \frac{U_{Q1}^2}{R_{Q1}} \\ \Rightarrow R_{Q1} &= \frac{U_{Q1}^2}{P_{Q1}} \end{aligned}$$

$$\begin{aligned} I_B &= \frac{I_{Q1}}{\beta \cdot \alpha} = \frac{430mA}{175} = 2,46mA \\ &= 2,46mA < 10mA \end{aligned}$$

$$\alpha = 0,5$$

$\hat{\alpha} = \text{doppelte Übersteuerung}$

$$R_B = \frac{U_{Hmin} - U_{BE}}{I_B} \approx 630\Omega = \frac{2,4V - 0,85V}{2,46mA}$$

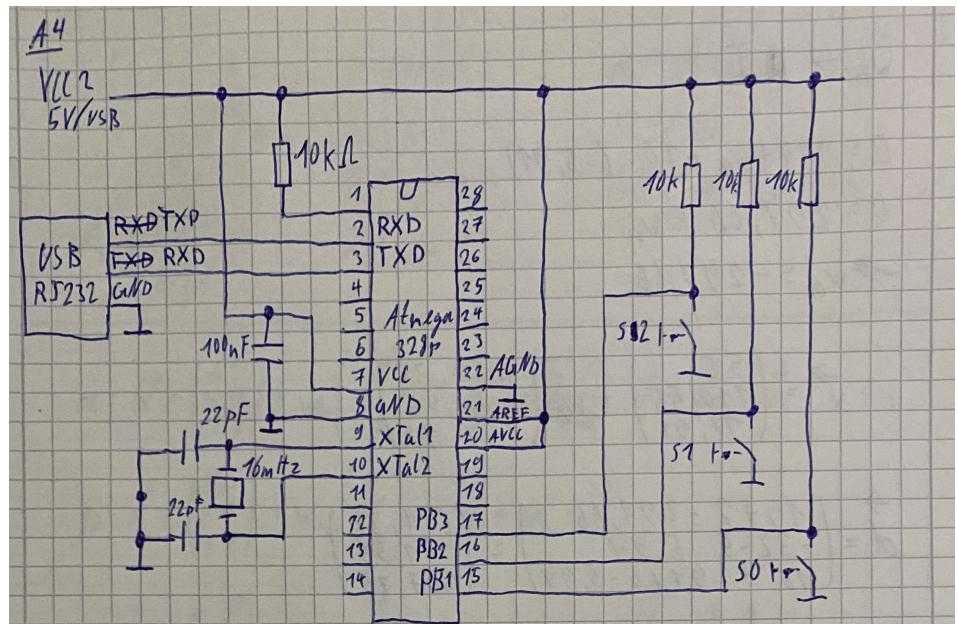
$$R_{Bmin} = \frac{U}{I_{Hmax}} = \frac{5V}{10mA} = 500\Omega$$

E12 Reihe: $|680\Omega \approx 630\Omega| > 500\Omega$

Ich entscheide mich für $R_B = 680\Omega$, durch die Übersteuerung ist der Strom nicht zu niedrig.

BRUNNEN

Abbildung 2: Aufgabe 3



- 16MHz Quarz zwischen XTAL1 (9) und XTAL2 (10).
- 22pF Kondensatoren zwischen jeweils Pin 9 und 10 und Ground.
- 10kΩ Widerstand zwischen 5V und Reset (1)
- 100nF Kondensator zwischen 5V und Ground des IC.
- AREF auf 5V
- RS232 an Serielle Kommunikation anschließen.

A5

(gezeichnet in A4)

Im Grundzustand liegt der jeweilige Pin auf High. Der 10kΩ Widerstand schützt vor einem Kurzschluss. Wenn ein Taster betätigt wird, liegt der MCV Pin auf Low.

Abbildung 3: Aufgabe 4 und 5

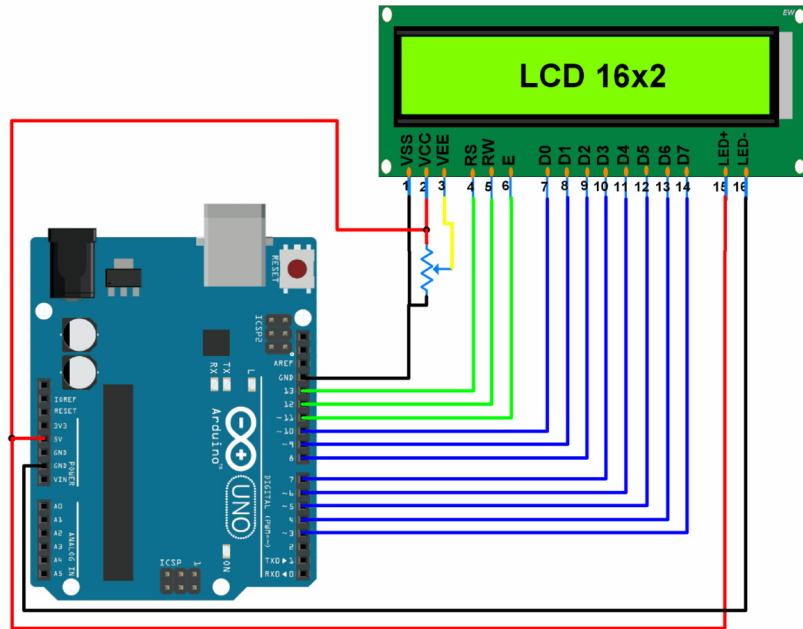


Abbildung 4: Schaltplan um LCD (mit HD44780) an Arduino anzuschließen

1.2 Aufgabe 6

In Abbildung 4 wie man ein LCD an einen Arduino anschließen kann. Die folgenden Pinbeschreibungen wurden aus dem Datenblatt des HD44780 entnommen.

Pin	MCU Pin	Funktionen der Ausgänge des HD44780
VSS	GND	niedriges Spannungspotential
VCC	5 V	positive Spannung für das LCD und den HD44780
VEE	[0 V; 5 V]	Kontrast – stellt ein wie stark sich die Kristalle drehen.
RS	PB5	0: Instruction 1: Data register
R/W	5 V	In diesem Fall wird nur geschrieben.
E	PB3	Enable – an ihm wird das Taktsignal angelegt.
D0 - D2	PB2 - PB0	8 Bit Dateneingang
D3 - D7	PD7 - PD3	8 Bit Dateneingang
LED+	5 V	Zuständig für die Spannungsversorgung der LEDs. Dabei ist darauf zu achten, dass ein Vorwiderstand in dem LCD Modul vorhanden ist.
LED-	GND	Hier muss das niedrige Spannungspotential angelegt werden.

1.3 Aufgabe 7

Nr	Fachkompetenz: Ich kann,	Tax moodle	Tätigkeitsbeschreibung: Ich habe,	T N W
0	für einen DC Motor (Lüfter) 12V / 600 W ein Rechts-/Linkslaufsteuerung mit Relais und galvanisch getrennt skizziert. VCC1	☺ ☺ 0,5	die Schaltungsvorlage mit allen benötigten Bauteilen und Verbindungen ergänzt.	
1	für ein Relais 5V / 2,5W eine Treiberstufe mit einem NPN Transistoren für einen TTL MC-PORT skizziert. VCC2	☺ ☺ 0,5	die Schaltungsvorlage mit allen benötigten Bauteilen und Verbindungen ergänzt.	
2	die Aufgabe 0 und 1 erklären.	☺ ☺ 0,5	die Schaltungen erklärt.	
3	die Aufgabe 1 <u>dimensionieren</u> .	☺ ☺ ☺ 3,0	die Schaltung für einen BC337/40 mit $U_{BE}=0,85V$, $B=350$, $U_{CESAT}=0,7V$, MC-PORT $U_{Hmin}=2,4V$, $I_{Hmax}=10mA$ dimensioniert.	
4	die Schaltungsvorlage mit einer kompletten MC-Basis-Schaltung (Quarz, Reset, ADC [AREF=VCC2], RS232/USB und VCC2)	☺ ☺ 2,0	die Schaltungsvorlage mit allen notwendigen Bauelementen + Werten inkl. Spannungszufuhr VCC2 USB → 5V ergänzt.	
5	3 Taster an einen MC in Negativlogik anschließen.	☺ ☺ 1,5	die Schaltungsvorlage aus dem Unterricht ergänzt und erklärt. (S0 = Aus, S1 = R, S2 = L)	
6	den 8 Bit Modus des HD44780 16x2 erklären.	☺ ☺ 2,5	die Schaltungsvorlage aus dem Unterricht ergänzt und erklärt.	
7	den 8 Bit Datenbus des HD44780, um eine 7 Seg. Anzeige mit 74HC573 ergänzen.	☺ ☺ 2,5	die Schaltungsvorlage analog zum Unterricht mit allen Bauteilen + Werten ergänzt und erklärt.	
8	ein MC Programm schreiben, das den Zustand der Taster aus A5 abfragt und per UART sendet. Grundsätzlich ist beim Programmstart die aktuelle Versionsnummer auszugeben.	☺ ☺ 2,0	das Basisprogramm ET_50_KL_01 im AVR-Studio (in C) mit dieser Funktion erstellt. BAUD-Rate = 19200 (ZWL und selbsterklärende Kommentare, Modularer Aufbau für alle Programme!!)	
9	das MC Programm aus A8 als Basis mit folgender Funktion ergänzen: Mit S1 wird das Relais R angesteuert, mit S2 wird das Relais L angesteuert, mit S0 werden alle Relais ausgeschaltet. Die Zustände der Relais werden gehalten und per UART gesendet.	☺ ☺ 2,0	das Basisprogramm ET_50_KL_01 zu ET_50_KL_02 kopiert und im AVR-Studio mit dieser Funktion ergänzt. Ausgaben sind mit Screenshots zu belegen.	
10	das Programm aus A9, um folgende Funktion ergänzen: empfängt das MC-Programm ein '0' → Motor ausschalten '1' → Motor Linkslauf einschalten 'r' → Motor Rechtslauf einschalten	☺ ☺ 2,0	das Basisprogramm ET_50_KL_02 zu ET_50_KL_03 kopiert und im AVR-Studio mit dieser Funktion ergänzt.	
11	das Programm aus A9, um folgende Funktion ergänzen: Der Zustand der Relais aus A10 auf Anfrage per RS232/USB mit einem 'i' beantwortet. ('L' → Links ; 'R' → Rechts ; '0' → aus)	☺ ☺ 2,0	das Basisprogramm ET_50_KL_03 zu ET_50_KL_04 kopiert und im AVR-Studio mit dieser Funktion ergänzt.	
12	die Initialisierung des LCD in A11 einbinden initLCD() und ausführlich erklären.	☺ ☺ 2,0	das Basisprogramm ET_50_KL_04 zu ET_50_KL_05 kopiert und im AVR-Studio mit dieser Funktion ergänzt.	
13	das Programm aus A12, um folgende Funktion ergänzen: Die Versionsnummer und der Motorzustand wird auf dem LCD	☺ ☺ 2,0	das Basisprogramm ET_50_KL_05 zu ET_50_KL_06 kopiert und im AVR-Studio mit dieser Funktion ergänzt.	

ET_50_KL_Maschinen_Transistor_Digitaltechnik_MC

	angezeigt.			
14	das Programm aus A13, um folgende Funktion ergänzen: Der Motorzustand wird auf der LED-Anzeige wie folgt dargestellt. 'o' = aus, 'l' = Linkslauf, 'r'=Rechtslauf	☺ ☺ 2,0	das Basisprogramm ET_50_KL_06 zu ET_50_KL_07 kopiert und im AVR-Studio mit dieser Funktion ergänzt.	
15	die Aufgabe 6.2.3.1 Nr. 3 lösen.	☺ ☺ 0,5	die Aufgabe ausführlich gelöst.	
16	die Aufgabe 6.10.1 Nr. 2, 5, 6 lösen.	☺ ☺ 1,5	die Aufgaben ausführlich gelöst.	
17	die Aufgaben 6.10.2 Nr. 1,2,3,4 lösen.	☺ ☺ 2,0	die Aufgaben ausführlich gelöst.	
19	die Aufgaben 6.10.3 Nr. 1,2,3,4,6 lösen.	☺ ☺ 2,5	die Aufgaben ausführlich gelöst.	
20	die Aufgaben 6.10.4 Nr. 1,3,4 lösen.	☺ ☺ 1,5	die Aufgaben ausführlich gelöst.	
21	die Aufgaben 6.10.5 Nr. 1,2,3 lösen.	☺ ☺ 1,5	die Aufgaben ausführlich gelöst.	
22	die Aufgaben 7.5.3 Nr. 1,2,3,4,7 lösen.	☺ ☺ 2,5	die Aufgaben ausführlich gelöst.	
23	die Aufgaben 7.5.4 Nr. 1,2,3,4 lösen.	☺ ☺ 2,0	die Aufgaben ausführlich gelöst.	
24	die Aufgaben 7.6.4 Nr. 1,2,3,4 lösen.	☺ ☺ 2,0	die Aufgaben ausführlich gelöst.	
25	die Aufgaben 7.7 Nr. 1,2,3,4 lösen.	☺ ☺ 2,0	die Aufgaben ausführlich gelöst.	
26	die Aufgaben 8.2 Nr. 1,2,3,4,6,7,8 lösen.	☺ ☺ 3,0	die Aufgaben ausführlich gelöst.	
27	die Aufgaben 8.3 Nr. 3 lösen.	☺ ☺ 1,0	die Aufgaben ausführlich gelöst.	
28	die Aufgaben 8.5 Nr. 1,5 lösen.	☺ ☺ 2,0	die Aufgaben ausführlich gelöst.	
29	die Aufgaben 8.6 Nr. 1,3 lösen.	☺ ☺ 2,0	die Aufgaben ausführlich gelöst.	

**Alle vorherigen Kann-Listen gelten als Grundlagen
und**

sind ebenfalls prüfungsrelevant.

**Es ist immer der vollständige Lösungsweg
inkl. Betriebsmittelkennzeichnung usw. anzugeben !!!**

Für die Bearbeitung sind das **Datenblatt ATMEGA328p.pdf** und **Mikrocontroller.net** wichtig.

<https://www.mikrocontroller.net/articles/AVR-Tutorial>

<https://www.mikrocontroller.net/articles/AVR-GCC-Tutorial>

Alle vorherigen Kann-Listen sind ebenfalls Gegenstand der Klausur.

Unterrichtsmitschriften, FKB, FRB und Projektvorträge, Videos zum Thema

<https://www.youtube.com/playlist?list=PLBF281451AA9A2E13> , <https://www.elektroniktutor.de> u.a.

☺ Reproduktion	☺ ☺ Reorganisation	☺ ☺ ☺ Anwendung	☺ ☺ ☺ ☺ Problemlösung
----------------	--------------------	-----------------	-----------------------

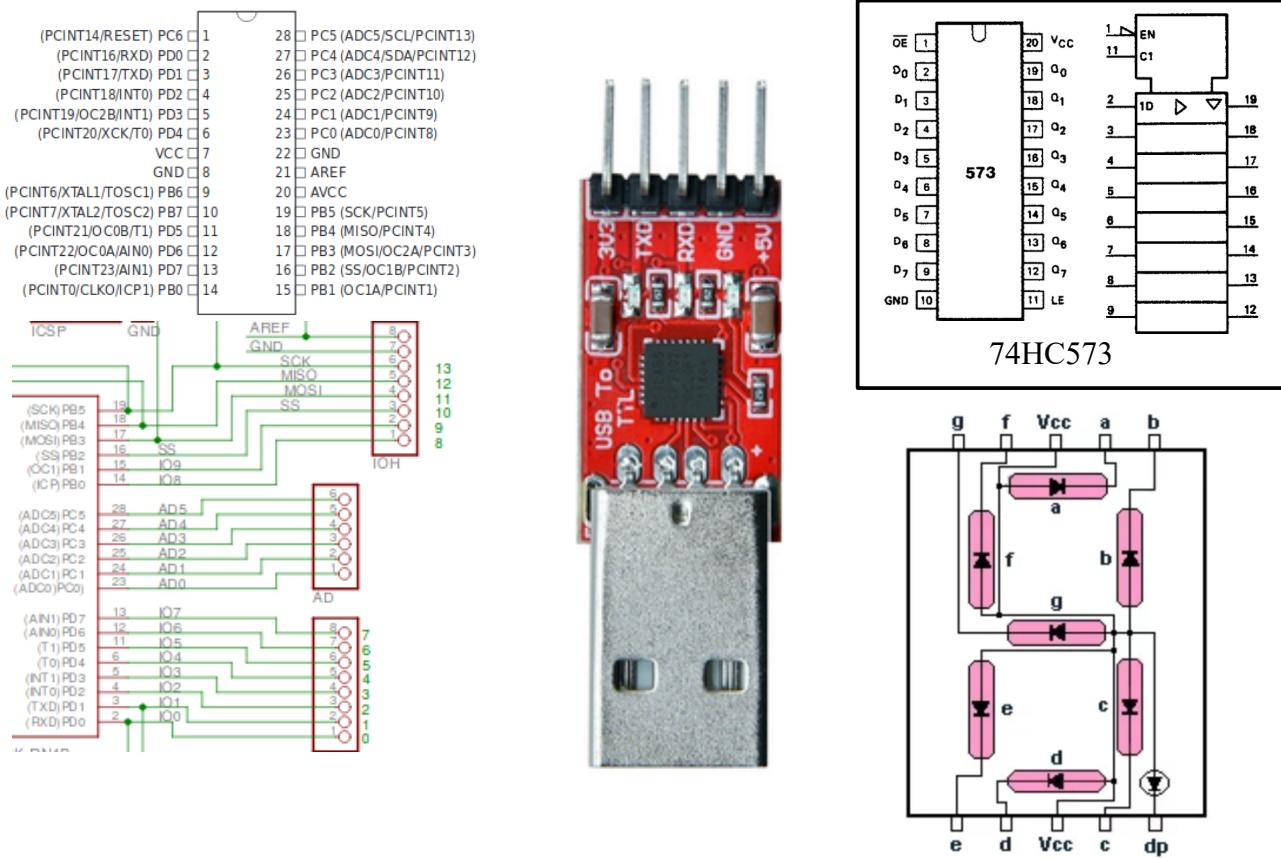
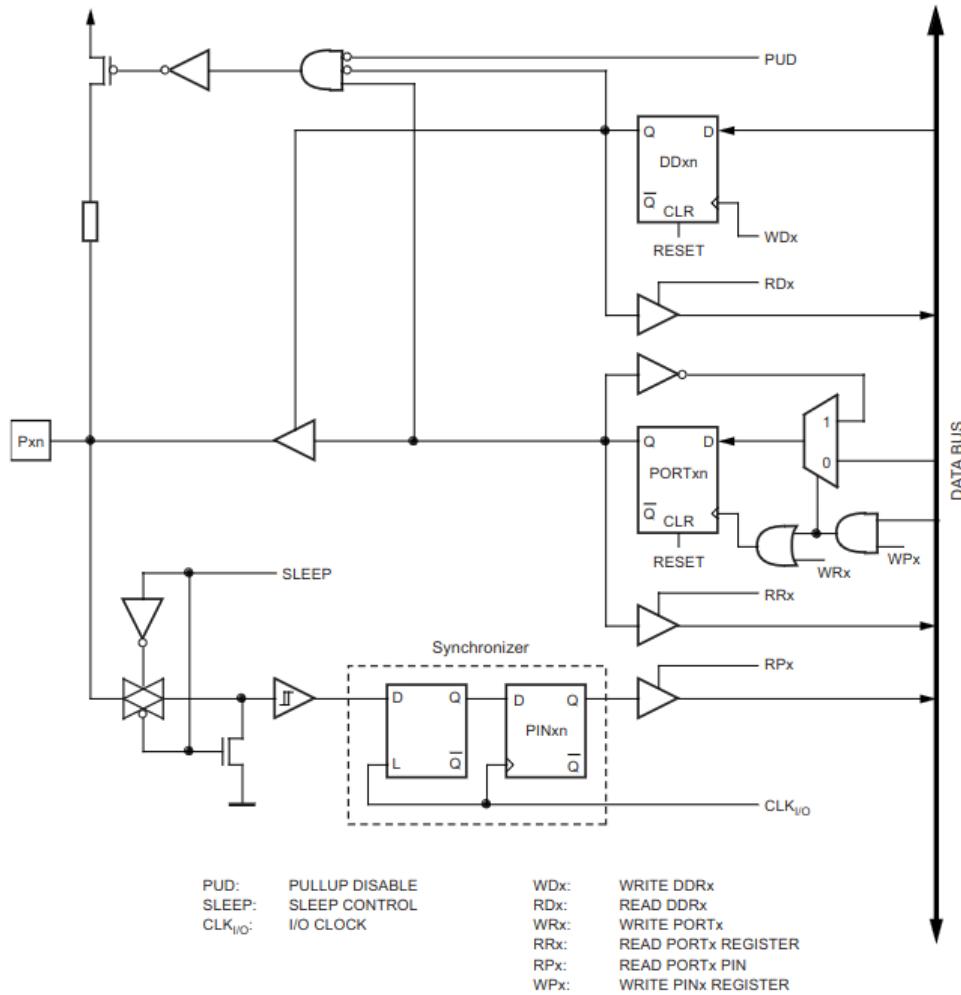
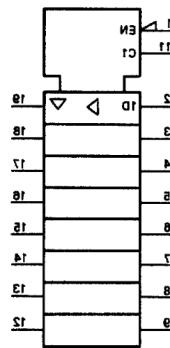
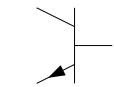
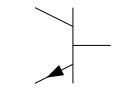
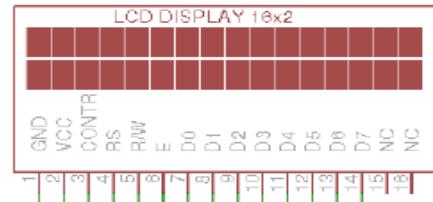
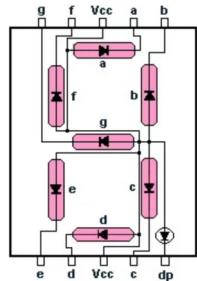


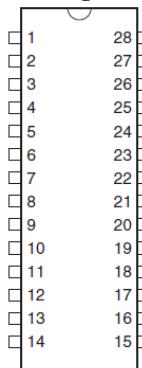
Figure 13-2. General Digital I/O⁽¹⁾



VCC 2
+5V /USB



Atmega328



Motor ZL

VCC 1
+12V
GND 1



GND 2
USB

