amwellius ...		1 minute ago
..		
06-lcd		25 minutes ago
images		2 minutes ago
README.md		1 minute ago

README.md

Lab 6: Samuel Košík

Link to my `Digital-electronics-2` GitHub repository:

(https://github.com/amwellius/Digital-electronics-2)

LCD display module

PREPARATION

LCD signal(s)	AVR pin(s)	Description
RS	PB0	Register selection signal. Selection between Instruction register (RS=0) and Data register (RS=1)
R/W	GND	Write/Read select
E	PB1	Signal Enable
D[3:0]	NOT Connected	Pins for Data

LCD signal(s)	AVR pin(s)	Description
D[7:4]	PD[7:4]	Pins for Data

What is the ASCII table? What are the codes/values for uppercase letters `A` to `Z`, lowercase letters `a` to `z`, and numbers `0` to `9` in this table?
=yes

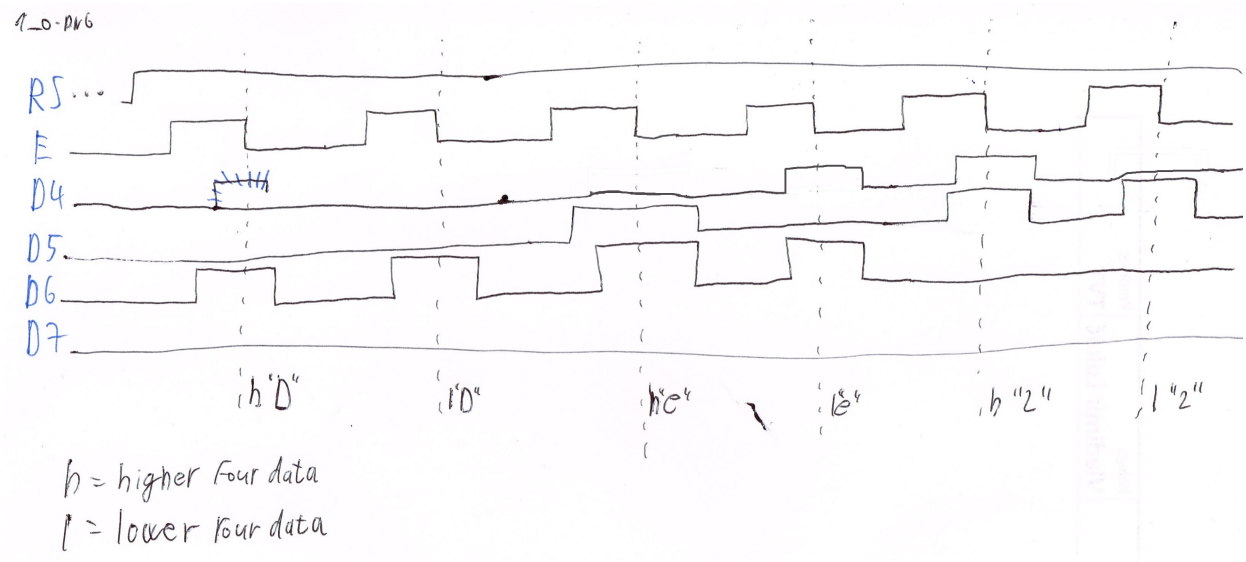
Char	Decimal	Hexadecimal
`A`	65	0x41
`B`	66	0x42
`Q`	81	0x51
`a`	97	0x61
`b`	98	0x62
`y`	121	0x79
`0`	48	0x30
`1`	49	0x31
`DEL`	127	0x7F
`>`	62	0x3E

PART 3 Table

Function name	Function parameters	Description	Example
`lcd_init`	`LCD_DISP_OFF` `LCD_DISP_ON`	Display off Display ON, Cursor Off Dipslay ON,	`lcd_init(LCD_DISP_OFF);`

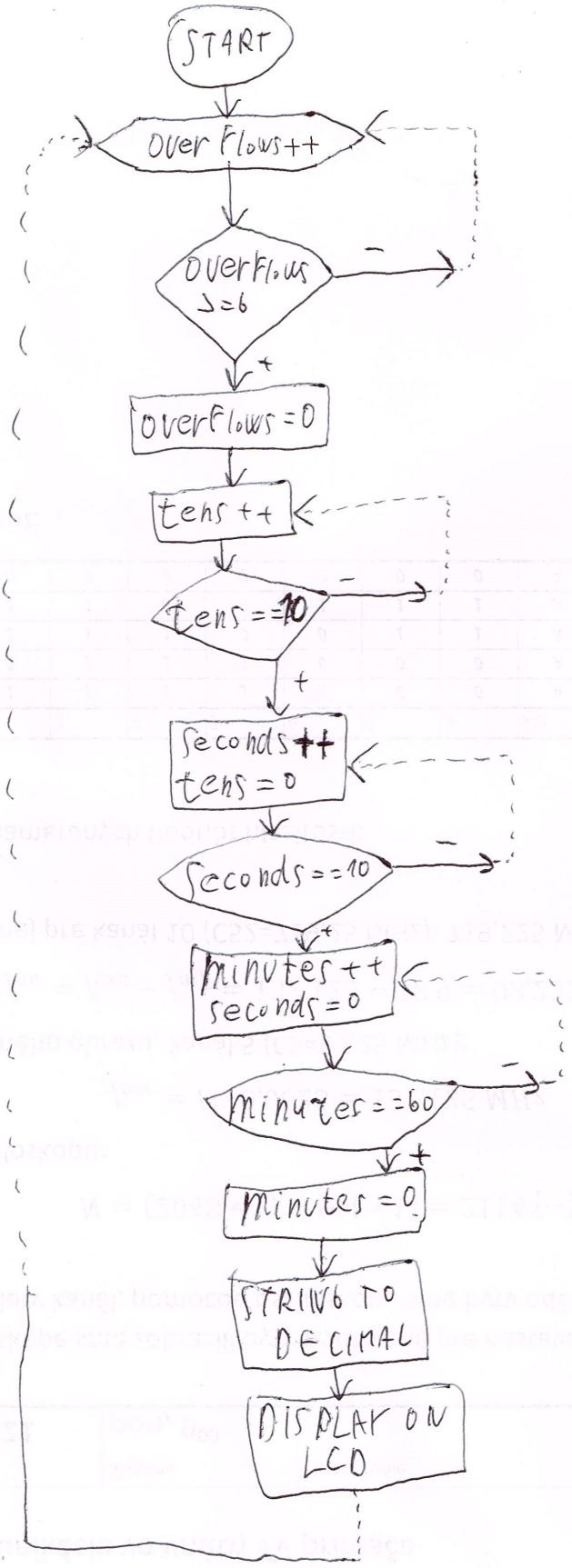
Function name	lcd_init LCD_DISP_ON_CURSOR LCD_DISP_ON_CURSOR_BLINK	Description Cursor ON Display ON, Cursor Flashing	Example
lcd_clrscr	nothing (void)	Clear display and set cursor to home position	lcd_clrscr();
lcd_gotoxy	uint8_t x uint8_t y	Set cursor to specified position	lcd_gotoxy(xpos,ypos)
lcd_putc	c	Display character at current cursor position	lcd_putc(c)
lcd_puts	s	Display string without auto linefeed	lcd_puts(s)
lcd_command	cmd	Send LCD controller instruction command	lcd_command(cmd)
lcd_data	data	Send data byte to LCD controller	lcd_data(data)

1. In your words, describe what ASCII table is.
    - ASCII = `American Standard Code For Information Interchange` . Tabulka, podla ktorej sa premienaju pismena, znaky, cisla do binarnej/hexadecimalnej sustavy. Ked chceme pri kodovani zobrazit znak, najdeme si v tabulke jeho hodnotu a tu zadame do kodu. Ten pri kompilovani zisti (podla ASCII tabulky) o ktory znak a ide a ten zobrazi.
2. (Hand-drawn) picture of time signals between ATmega328P and LCD keypad shield (HD44780 driver) when transmitting three character data `De2` .



Stopwatch

1. Flowchart figure for `TIMER2_OVF_vect` interrupt service routine which overflows every 16 ms but it updates the stopwatch LCD approximately every 100 ms (6 x 16 ms = 100 ms). Display tenths of a second and seconds `minutes:seconds.tenths` . Let the stopwatch counts from `00:00.0` to `59:59.9` and then starts again. The image can be drawn on a computer or by hand. Use clear descriptions of the individual steps of the algorithms.



Custom characters

1. Code listing with syntax highlighting of two custom character definition:

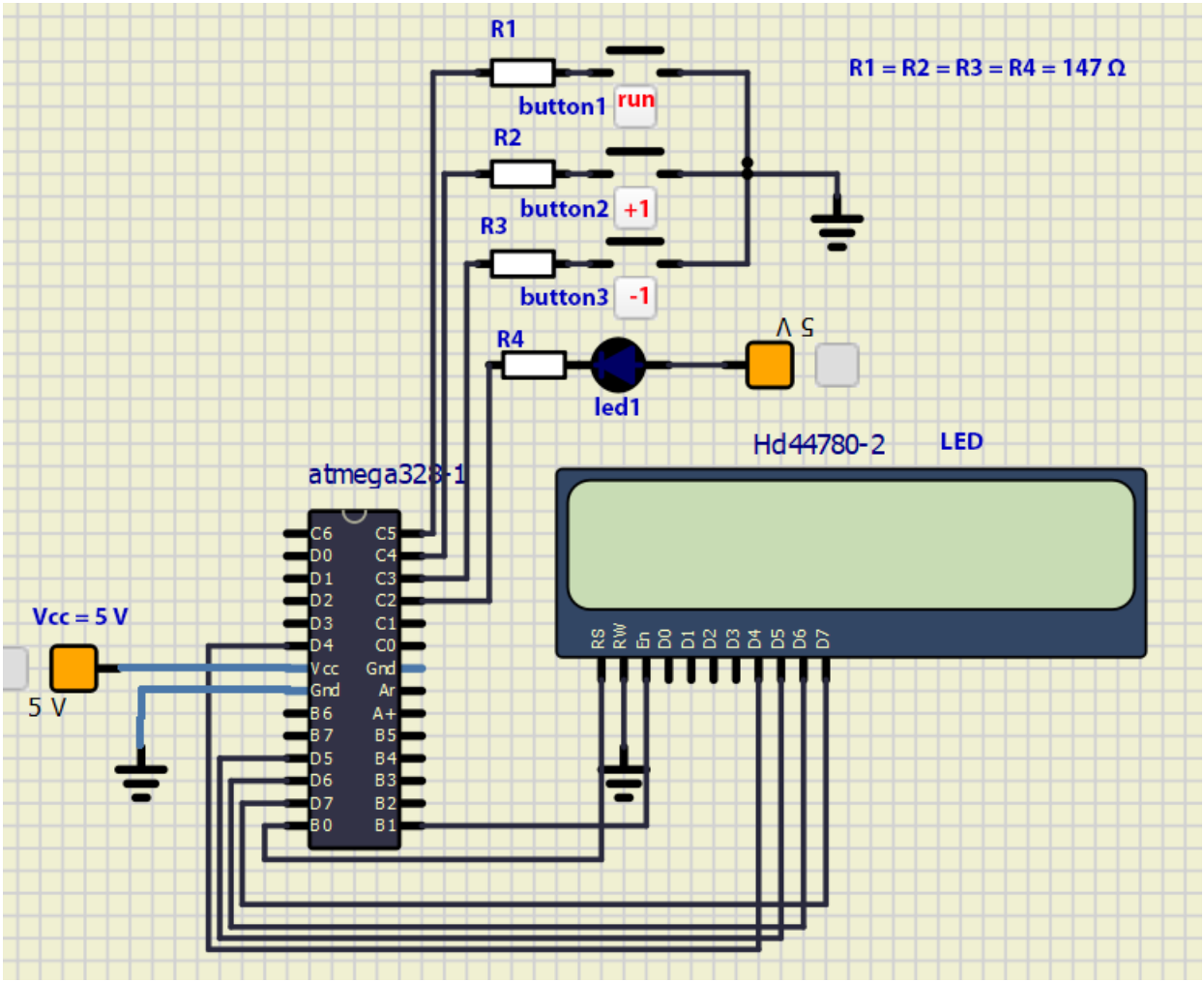
```
uint8_t customChar2[8] = {
    0b11111,
    0b10010,
    0b10100,
    0b11000,
    0b10100,
    0b10010,
    0b10001,
    0b11111
};

uint8_t customChar1[8] = {
    0b10011,
    0b10111,
    0b11011,
    0b11000,
    0b01000,
    0b10111,
    0b11001,
    0b00011
};
```

Kitchen alarm

Consider a kitchen alarm with an LCD, one LED and three push buttons: start, +1 minute, -1 minute. Use the +1/-1 minute buttons to increment/decrement the timer value. After pressing the Start button, the countdown starts. The countdown value is shown on the display in the form of mm.ss (minutes.seconds). At the end of the countdown, the LED will start blinking.

1. Scheme of kitchen alarm; do not forget the supply voltage. The image can be drawn on a computer or by hand. Always name all components and their values.


R1
R1 = R2 = R3 = R4 = 147 Ω
button1
run
R2
button2
+1
R3
button3
-1
R4
led1
Hd44780-2
LED
atmega328-1
Vcc = 5 V
5 V
C6
D0
D1
D2
D3
D4
Vcc
Gnd
B6
B7
D5
D6
D7
B0
C5
C4
C3
C2
C1
C0
Gnd
Ar
A+
B5
B4
B3
B2
B1
RS
RW
En
D0
D1
D2
D3
D4
D5
D6
D7