

1.

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	Read/Write selection
E	PB1	loads the data into the HD44780 on the falling edge
D[3:0]	Not connected	Lower nibble of 8 bit data
D[7:4]	PD7 to PD4	Higher nibble of 8 bit data

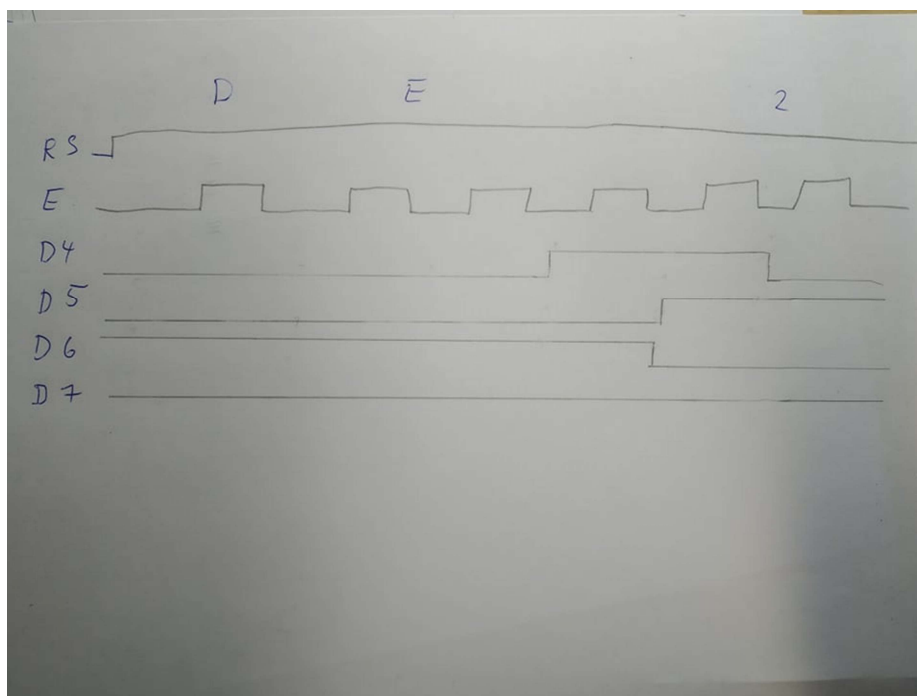
ASCII je americký kód pro výměnu informací. Je to standart přiřazení kódu k symbolu.

A-Z je kód 0x41 až 0x5a

a-z je kód 0x61 až 0x7a

0-9 je kód 0x30 až 0x39

2.



3.

```
ISR(TIMER2_OVF_vect)
{
    static uint8_t number_of_overflows = 0;
    static uint8_t tens = 0;          // Tenths of a second
    static uint8_t secs = 0;          // Seconds
    static uint8_t minutes = 0;       // minutes
    static uint16_t square_minutes = 0; //square minutes
    char lcd_string[2] = " ";        // String for converting numbers by itoa()

    number_of_overflows++;
    if (number_of_overflows >= 6)
    {
        // Do this every 6 x 16 ms = 100 ms
        number_of_overflows = 0;

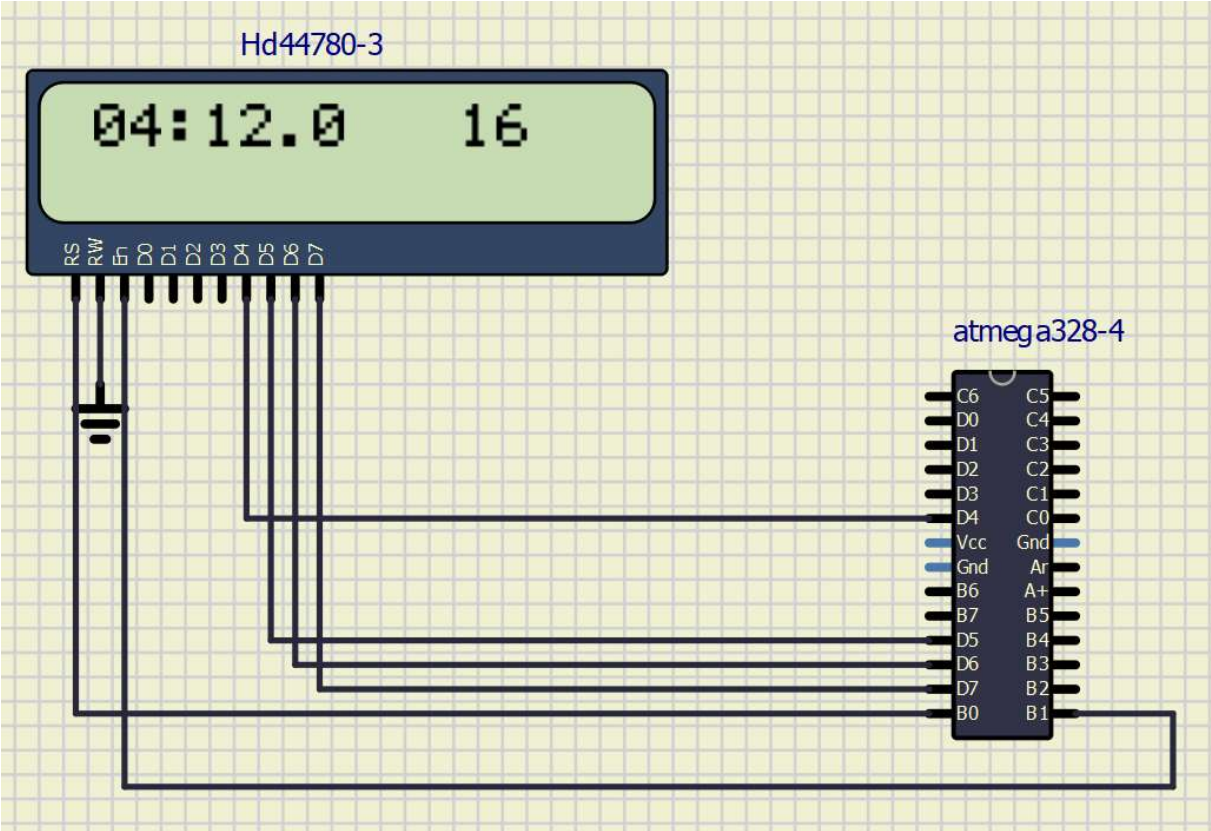
        tens++;
        if (tens > 9)
        {
            tens = 0;
            secs++;
            if (secs > 59)
            {
                secs = 0;
                lcd_gotoxy(4, 0);
                lcd_puts("00");
                minutes++;
                if (minutes > 59)
                {
                    minutes = 0;
                    lcd_gotoxy(1, 0);
                    lcd_puts("00");

                    lcd_gotoxy(12, 0); //clear last number for next square value
                    lcd_puts(" ");

                    square_minutes = minutes * minutes;
                    itoa(square_minutes, lcd_string, 10); // Convert decimal value to string
                    lcd_gotoxy(11, 0);
                    lcd_puts(lcd_string); //show square minutes
                }
            }
        }
        itoa(tens, lcd_string, 10); // Convert decimal value to string
        lcd_gotoxy(7, 0);
        lcd_puts(lcd_string);

        itoa(secs, lcd_string, 10); // Convert decimal value to string
        if (secs > 9)
        {
            lcd_gotoxy(4, 0);
            lcd_puts(lcd_string);
        }
        else
        {
            lcd_gotoxy(5, 0);
            lcd_puts(lcd_string);
        }

        itoa(minutes, lcd_string, 10); // Convert decimal value to string
        if (minutes > 9)
        {
            lcd_gotoxy(1, 0);
            lcd_puts(lcd_string);
        }
        else
        {
            lcd_gotoxy(2, 0);
            lcd_puts(lcd_string);
        }
    }
}
```



4.

```
ISR(TIMER0_OVF_vect)
{
    static uint8_t symbol = 0;
    static uint8_t position = 0;

    lcd_gotoxy(1 + position, 1);
    lcd_putc(symbol);

    symbol++;
    if (symbol > 5)
    {
        symbol = 0;
        position++;
    }

    if (position > 9)
    {
        position = 0;
        lcd_gotoxy(1 + position, 1);
        lcd_puts("      ");
    }
}
```

