

1) Preparation tasks

Table with voltage divider, calculated and measured ADC values for all buttons

Push button	PC0 [A0] voltage (calculated)	PC0 [A0] voltage (measured)	ADC value (calculated)	ADC value (measured)
Right	0 V	0 V	0	0
Up	0,495 V	0,49 V	101	101
Down	1,202 V	1,20 V	246	245
Left	1,970 V	1,96 V	403	402
Select	3,182 V	3,18 V	651	650
none	5 V	4,99 V	1023	1022

2) ADC

Listing of ADC_vect from file main.c

```

ISR(ADC_vect)
{
    uint16_t value = ADC;
    char lcd_string[8] = "      ";

    // Clear decimal and hex positions
    lcd_gotoxy(8, 0);
    lcd_puts(lcd_string);

    // Print ADC value on LCD in decimal
    itoa(value, lcd_string, 10);
    lcd_gotoxy(8, 0);
    lcd_puts(lcd_string);

    if (value < 700)
    {
        // Send data through UART
        uart_puts("ADC value in decimal: ");
        uart_puts(lcd_string);
        uart_puts("\r\n");
    }

    // Print ADC value on LCD in hex
    itoa(value, lcd_string, 16);
    lcd_gotoxy(13, 0);
    lcd_puts(lcd_string);

    // Clear key positions
    lcd_gotoxy(8, 1);
    lcd_puts(" ");
    lcd_gotoxy(8, 1);

    // Print key
    if (value > 1000)
    {
        lcd_puts("NONE");
    }

    if (value < 70)
    {
        lcd_puts("RIGHT");
    }

    if (value > 70 && value < 200)
    {
        lcd_puts("UP");
    }

    if (value > 200 && value < 370)
    {
        lcd_puts("DOWN");
    }

    if (value > 370 && value < 600)
    {
        lcd_puts("LEFT");
    }
}

```

```
if (value > 600 && value < 1000)
{
    lcd_puts("SELECT");
}

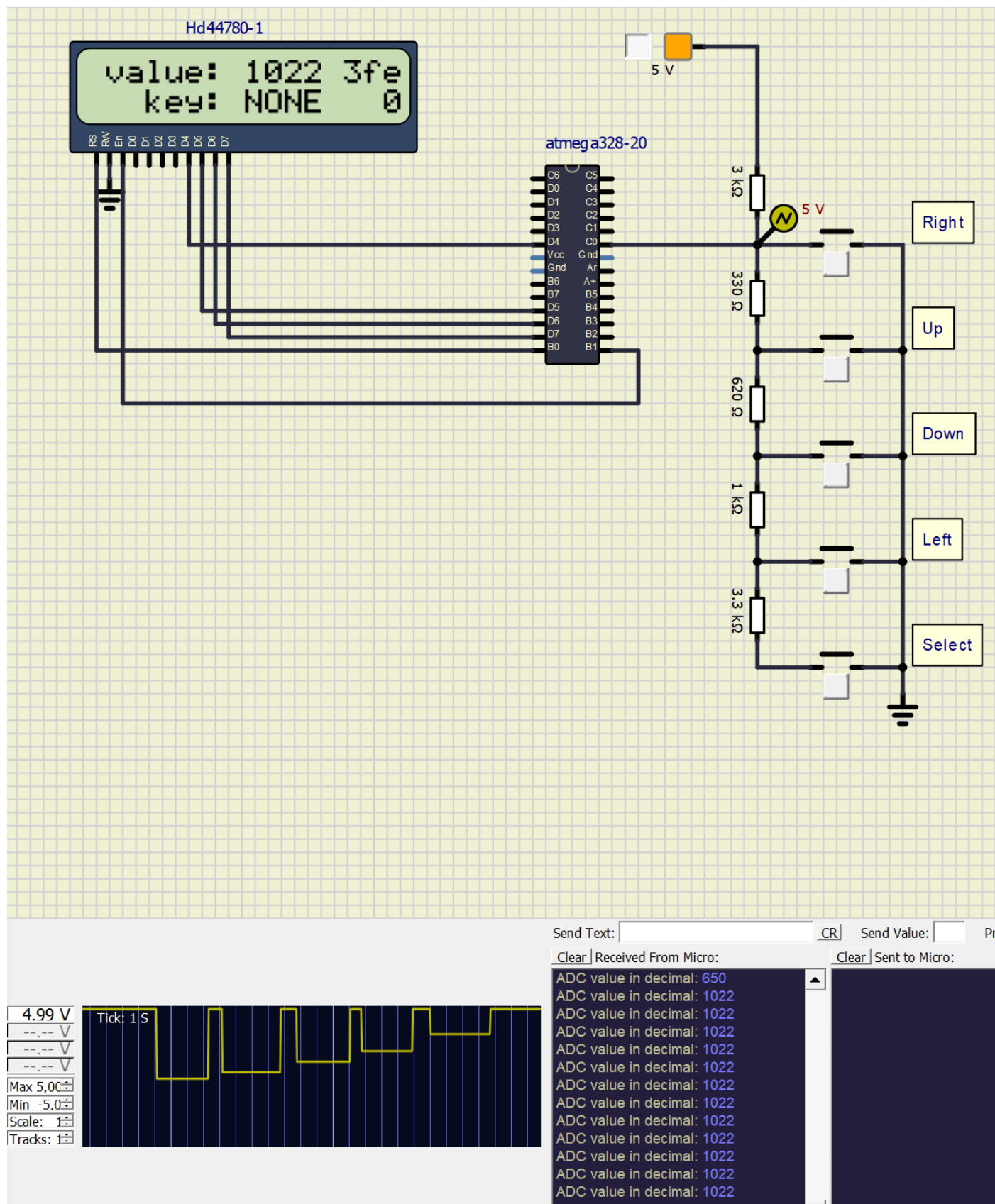
// Code for calculating and displaying parity bit
if (value > 1000)
{
    int parity = 0;
    itoa(parity, lcd_string, 10);
    lcd_gotoxy(15,1);
    lcd_puts(lcd_string);
}
else
{
    int memory[32];
    int i = 0, j, parity = 0;

    while (value > 0)
    {
        memory[i] = value % 2;
        value = value/2;
        i++;
    }

    for (j = i - 1; j > 0; j--)
    if(memory[j] == 1)
    {
        parity++;
    }
    else
    {
        parity = 0;
    }

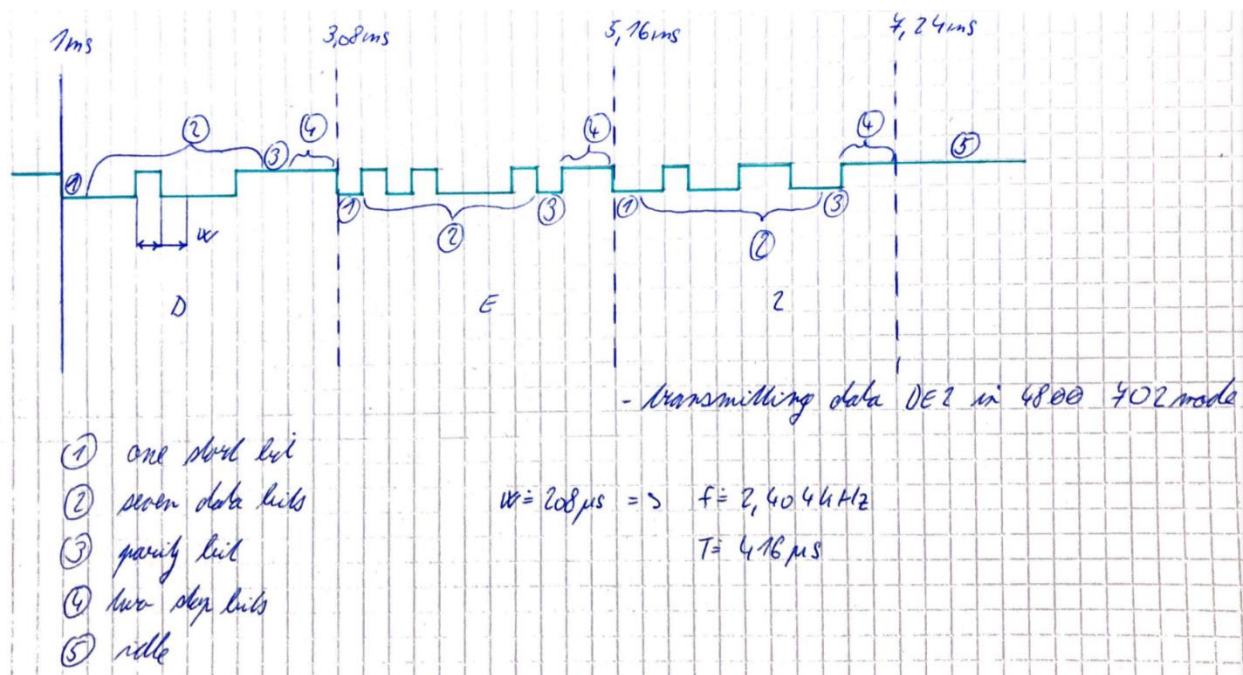
    itoa(parity, lcd_string, 10);
    lcd_gotoxy(15,1);
    lcd_puts(lcd_string);
}
}
```

Screenshot of SimulIDE circuit when "Power Circuit" is applied



3) UART

- transmitting data DE2 in 4800 702 mode



Listing of code for calculating and displaying parity bit from file main.c

```
// Code for calculating and displaying parity bit
if (value > 1000)
{
    int parity = 0;
    itoa(parity, lcd_string, 10);
    lcd_gotoxy(15,1);
    lcd_puts(lcd_string);
}
else
{
    int memory[32];
    int i = 0, j, parity = 0;

    while (value > 0)
    {
        memory[i] = value % 2;
        value = value/2;
        i++;
    }

    for (j = i - 1; j > 0; j--)
    {
        if(memory[j] == 1)
        {
            parity++;
        }
        else
        {
            parity = 0;
        }
    }

    itoa(parity, lcd_string, 10);
    lcd_gotoxy(15,1);
    lcd_puts(lcd_string);
}
```