

Lab 7: Tomáš Kříčka

My repository

<https://github.com/TomasKricka/Digital-electronic-2>

Analog-to-Digital Conversion

1. Complete table with voltage divider, calculated, and measured ADC values for all five push buttons.

Push button	PC0[A0] voltage	ADC value (calculated)	ADC value (measured in sumulIDE)
Right	0 V	0	0
Up	0.495 V	101	101
Down	1.203 V	246	245
Left	1.969 V	403	402
Select	3.181 V	651	650
none	5 V	1023	1022

2. Code listing of ACD interrupt service routine for sending data to the LCD/UART and identification of the pressed button. Always use syntax highlighting and meaningful comments:

```
/* *****  
 * Function: ADC complete interrupt  
 * Purpose: Display value on LCD and send it to UART.  
 * ***** */  
ISR(ADC_vect)  
{  
    // WRITE YOUR CODE HERE  
    uint16_t value = 0;  
    char lcd_string[4]= "  ";  
  
    value = ADC;  
  
    itoa(ADC, lcd_string, 10);  
    lcd_gotoxy(8, 0);  
    lcd_puts("  ");  
    uart_puts(lcd_string);  
    uart_puts("\r\n");  
    lcd_gotoxy(8, 0);  
    lcd_puts(lcd_string);  
  
    itoa(ADC, lcd_string, 16);  
    lcd_gotoxy(13, 0);  
    lcd_puts("  ");
```

```
    uart_puts(lcd_string);
    uart_puts("\r\n");
    uart_puts("\r\n");
    lcd_gotoxy(13, 0);
    lcd_puts(lcd_string);

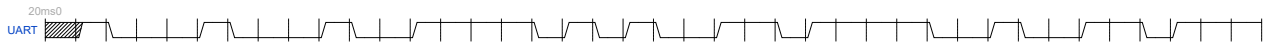
    lcd_gotoxy(8, 1);
    lcd_puts(" ");
    lcd_gotoxy(8, 1);

    if(ADC > 1010 )
    {
        lcd_puts("none");
        uart_puts("none");
    }
    else if (ADC > 590 && ADC < 690)
    {
        lcd_puts("select");
        uart_puts("select");
    }
    else if (ADC > 300 && ADC < 460)
    {
        lcd_puts("left");
        uart_puts("left");
    }
    else if (ADC > 190 && ADC < 300)
    {
        lcd_puts("down");
        uart_puts("down");
    }
    else if (ADC > 50 && ADC < 150)
    {
        lcd_puts("up");
        uart_puts("up");
    }
    else if (ADC < 20)
    {
        lcd_puts("right");
        uart_puts("right");
    }

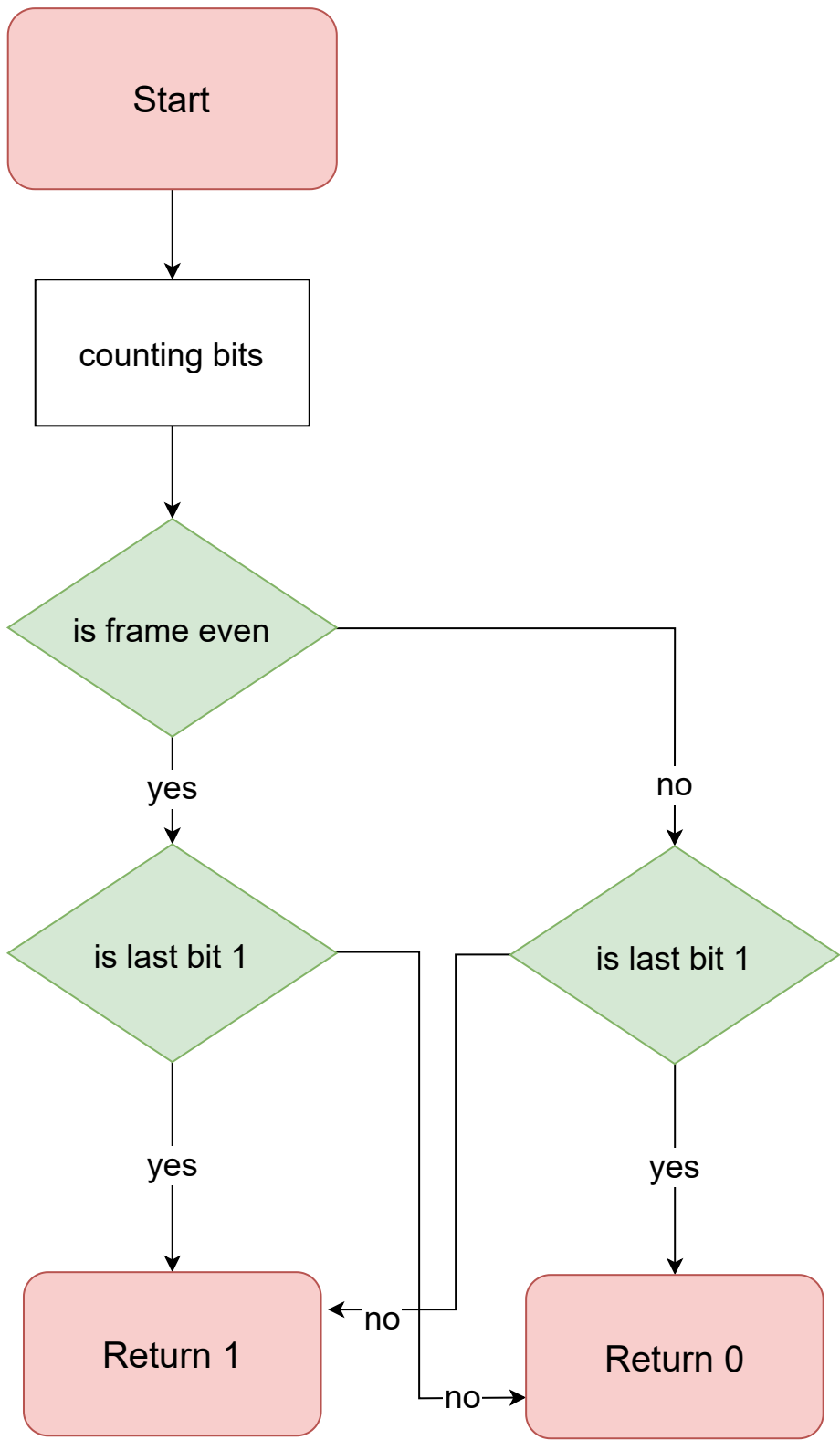
    uart_puts("\r\n");
}
```

UART communication

1. (Hand-drawn) picture of UART signal when transmitting three character data **De2** in 4800 7O2 mode (7 data bits, odd parity, 2 stop bits, 4800 Bd).



2. Flowchart figure for function `uint8_t get_parity(uint8_t data, uint8_t type)` which calculates a parity bit of input 8-bit `data` according to parameter `type`. The image can be drawn on a computer or by hand. Use clear descriptions of the individual steps of the algorithms.



Temperature meter

Consider an application for temperature measurement and display. Use temperature sensor [TC1046](#), LCD, one LED and a push button. After pressing the button, the temperature is measured, its value is displayed on the

1. Scheme of temperature meter. The image can be drawn on a computer or by hand. Always name all components and their values.

