Lab 4: Jiří Tomešek

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

https://github.com/xtomes07/Digital-electronics-2

Overflow times

1. Complete table with overflow times.

Module	Number of bits	1	8	32	64	128	256	1024
Timer/Counter0	8	16u	128u	--	1024 us	--	4.096 ms	16.384 ms
Timer/Counter1	16	4.096 ms	32.768 ms	--	262.144 ms	--	1.048576 s	4.194304 s
Timer/Counter2	8	16u	128u	512us	1024 us	2.048ms	4.096 ms	16.384 ms

Timer library

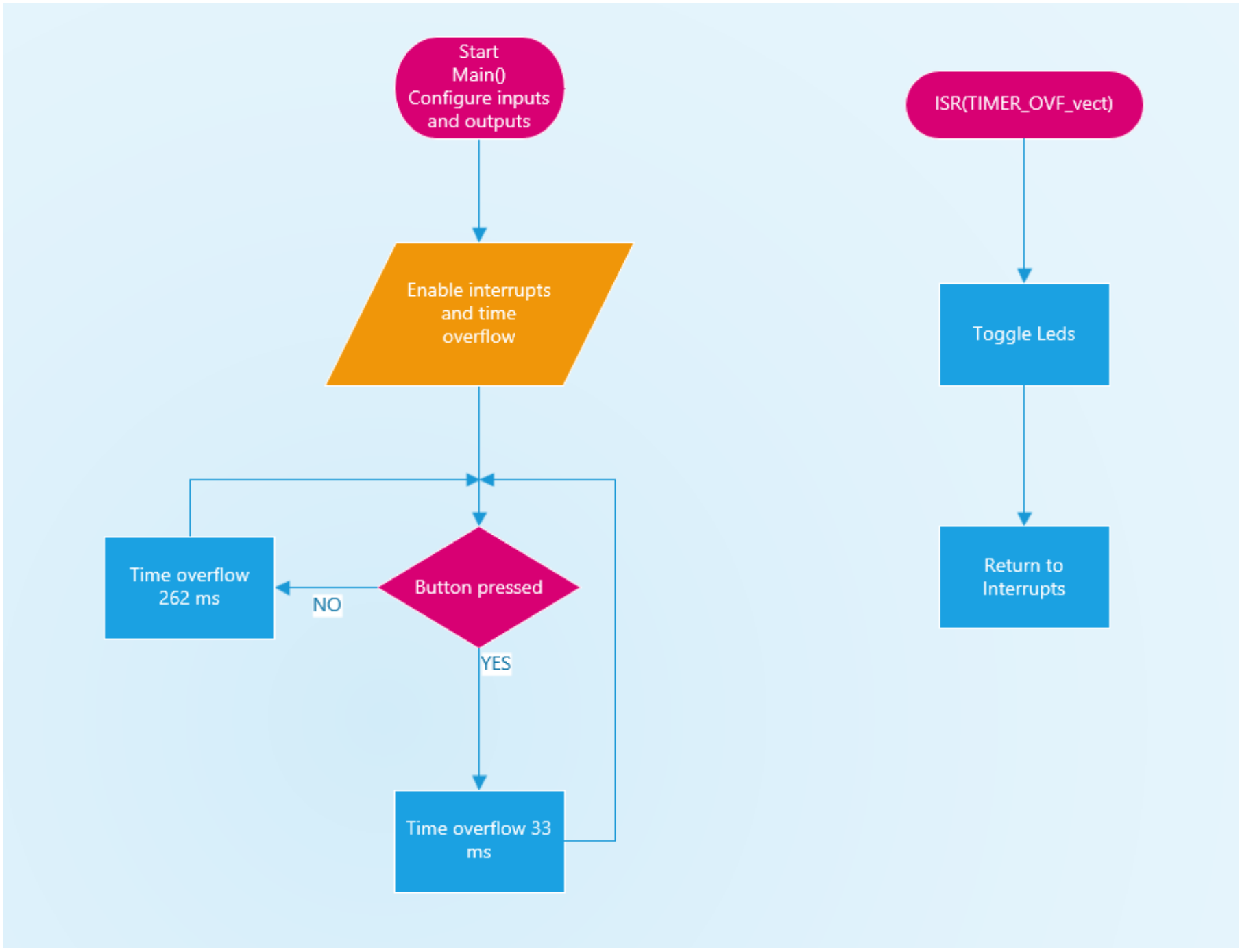
1. In your words, describe the difference between common C function and interrupt service routine.

   - Function - generický blok kódu, který budem v našem kódu víckrát používat. Tento blok lze volat z jakékoli části programu.
   - Interrupt service routine - funkce, na kterou skočíme, když dojde k přerušení. Hlavní program je zastaven a poté, co je ISR spuštěn, pokračuje ze stejného bodu, kde byl zastaven

2. Part of the header file listing with syntax highlighting, which defines settings for Timer/Counter0:

```
/**
 * @name  Definitions of Timer/Counter0
 * @note  F_CPU = 16 MHz
 */
#define TIM0_stop()                         TCCR0B &= ~((1<<CS02) | (1<<CS01) | (1<<CS00));
/** @brief Set overflow 16us, prescaler 001 --> 1 */
#define TIM0_overflow_16us()                TCCR0B &= ~((1<<CS02) | (1<<CS01)); TCCR0B |= (1<<CS00);
/** @brief Set overflow 128us, prescaler 010 --> 8 */
#define TIM0_overflow_128us()               TCCR0B &= ~((1<<CS02) | (1<<CS00)); TCCR0B |= (1<<CS01);
/** @brief Set overflow 1024 us, prescaler 011 --> 64 */
#define TIM0_overflow_1024us()              TCCR0B &= ~(1<<CS02); TCCR0B |= (1<<CS01) | (1<<CS00);
/** @brief Set overflow 4096us, prescaler 100 --> 256 */
#define TIM0_overflow_4096us()              TCCR0B &= ~((1<<CS01) | (1<<CS00)); TCCR0B |= (1<<CS02);
/** @brief Set overflow 16384 us, prescaler // 101 --> 1024 */
#define TIM0_overflow_16384us()             TCCR0B &= ~(1<<CS01); TCCR0B |= (1<<CS02) | (1<<CS00);
/** @brief Enable overflow interrupt, 1 --> enable */
#define TIM0_overflow_interrupt_enable()    TIMSK0 |= (1<<TOIE0);
/** @brief Disable overflow interrupt, 0 --> disable */
#define TIM0_overflow_interrupt_disable()   TIMSK0 &= ~(1<<TOIE0);
```

3. Flowchart figure for function `main()` and interrupt service routine `ISR(TIMER1_OVF_vect)` of application that ensures the flashing of one LED in the timer interruption. When the button is pressed, the blinking is faster, when the button is released, it is slower. Use only a timer overflow and not a delay library.



Knight Rider

1. Scheme of Knight Rider application with four LEDs and a push button, connected according to Multi-function shield. Connect AVR device, LEDs, resistors, push button, and supply voltage. The image can be drawn on a computer or by hand. Always name all components and their values!