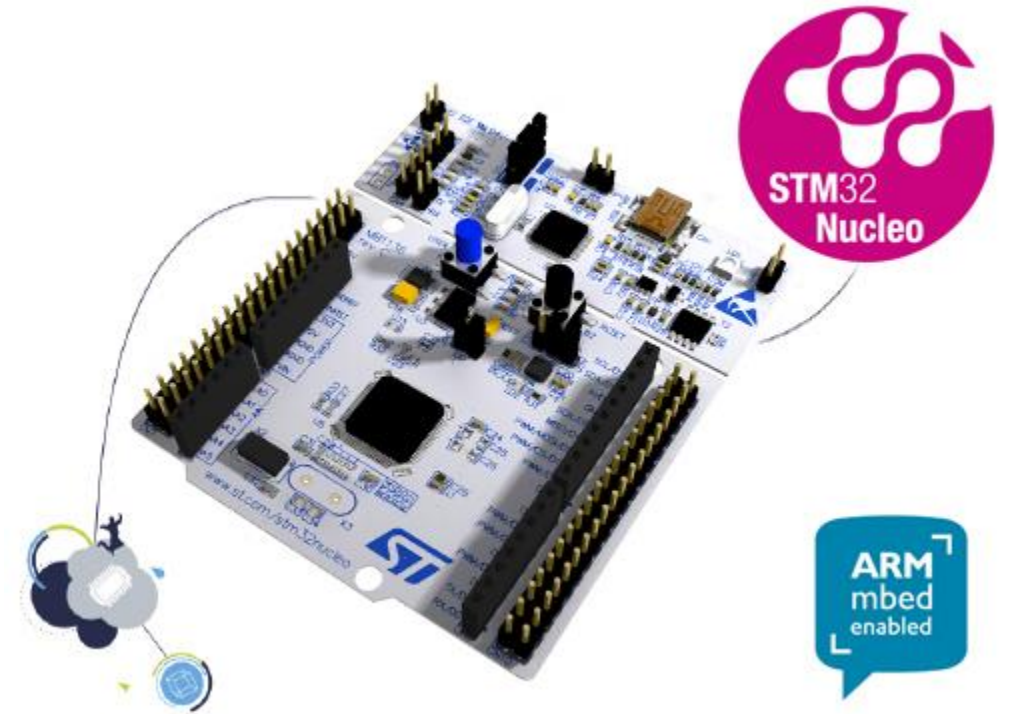


# Pokročilé asemblery

Cvičení 2, 2017

- Vývojová platforma s MCU od firmy ST Microelectronics.
- Pracovní frekvence až 180MHz.
- Spousta periferních zařízení uvnitř procesoru (I2C, SPI, Timers, USART, AD, DA,...).



- Vývojová prostředí: Atolic Studio (Eclipse), STM32 Workbench (Eclipse), emBitz (Code Blocks).
- Možnost programovat jak v jazyce C, tak i v jazyce symbolických instrukcí pro ARM zařízení.
- Prostředí k dnešnímu cvičení: [www.fit.vutbr.cz/~igoldmann/ARM\\_tools.zip](http://www.fit.vutbr.cz/~igoldmann/ARM_tools.zip)

- Podmíněné instrukce
- Návrat z funkce nastavením PC
- Vyzkoušejte instrukce: LDR, STR, MOV, PUSH, POP, ADD,...

- Cílem mapování je dostat se k adresám hodnot se kterými pracují periferní zařízení.
- Příklad mapování pro GPIO:

STM32F40x register boundary addresses

Bus	Boundary address	Peripheral
AHB1	0x4002 2000 - 0x4002 23FF	GPIOI
	0x4002 1C00 - 0x4002 1FFF	GPIOH
	0x4002 1800 - 0x4002 1BFF	GPIOG
	0x4002 1400 - 0x4002 17FF	GPIOF
	0x4002 1000 - 0x4002 13FF	GPIOE
	0x4002 0C00 - 0x4002 0FFF	GPIOD
	0x4002 0800 - 0x4002 0BFF	GPIOC
	0x4002 0400 - 0x4002 07FF	GPIOB
	0x4002 0000 - 0x4002 03FF	GPIOA

Source: <https://bluetechs.wordpress.com/zothers/x/gpio/>

```
typedef struct
{
    __IO uint32_t MODER; mode register, Address offset: 0x00
    __IO uint32_t OTYPER; output type register, Address offset: 0x04
    __IO uint32_t OSPEEDR; output speed register, Address offset: 0x08
    __IO uint32_t PUPDR; pull-up/pull-down register, Address offset: 0x0C
    __IO uint32_t IDR; input data register, Address offset: 0x10
    __IO uint32_t ODR; output data register, Address offset: 0x14
    __IO uint16_t BSRRL; < bit set/reset low register, Address offset: 0x18
    __IO uint16_t BSRRH; bit set/reset high register, Address offset: 0x1A
    __IO uint32_t LCKR; configuration lock register, Address offset: 0x1C
    __IO uint32_t AFR[2]; alternate function registers, Address offset: 0x20-0x24
} GPIO_TypeDef;
```

- Adresu lze nahrát do libovolného registru pomocí instrukce ldr, např. **ldr r0, =0x4002000**

- Stáhnout si pack nástrojů pro STM32  
Spustit **emBitz**  
**Projekt STM32\_F411\_emu** – slouží k emulaci, kde si můžete vyzkoušet použití ARM instrukcí  
**Projekt STM32\_F401RE, Projekt STM32\_F446RE** – k vytvoření ELF souboru pro nahrání na desku Nucleo
- Otevřít projekty a vyzkoušet překlad.
- Seznámit se kódem `main.c` a `tone_generator.s`
- Přehledný popis ARM instrukcí: <https://www.root.cz/clanky/instrukcni-sada-mikroprocesoru-arm/>

### Bodovaný úkol:

1. Rozblikat LED diodu na STM32 Kitu
2. Využít tlačítko pro rozsvícení LED diody
3. Generování tónu

<http://www.coranac.com/tonc/text/asm.htm>,

[https://www.csie.ntu.edu.tw/~cyu/courses/assembly/08fall/lectures/handouts/lec08\\_ARMisa\\_4up.pdf](https://www.csie.ntu.edu.tw/~cyu/courses/assembly/08fall/lectures/handouts/lec08_ARMisa_4up.pdf)