

Programujeme STM32

zdolejte jednočipy profesionálů

Ing. Vojtěch Skřivánek

```
RCC_OscInitTypeDef RCC_OscInitStruct = {0};
```

```
RCC_ClkInitTypeDef RCC_ClkInitStruct = {0};
```

```
/** Configure the main internal regulator output voltage
```

```
*/
```

```
HAL_PWR_VOLTAGESCALING_CONFIG(&PWR_InitStruct);
```

```
/** Initializes the CPU, AHB
```

```
*/
```

```
RCC_C
```

```
RCC_Osc
```

```
RCC_Osc
```

```
RCC_OscIn
```

```
if (HAL_RCC
```

```
{
```

```
Error_Handler(
```

```
}
```

```
/** Initializes the C
```

Za podpory firem

SIEMENS a



STMicroelectronics

Programujeme STM32

zdolejte jednočipy profesionálů

Ing. Vojtěch Skřivánek

Vydání první 2021

© Ing. Vojtěch Skřivánek

© Mgr. Tomáš Zahradníček - TZ-one

ISBN: 978-80-7539-115-5 (ePub verze)

ISBN: 978-80-7539-116-2 (mobi verze)

ISBN: 978-80-7539-117-9 (PDF verze)

Obsah

1	Úvod.....	7
1.1	Motivace knihy	7
1.2	Struktura knihy	7
1.3	Fonty textu	8
2	Vývojové nástroje	9
2.1	Vývojová deska.....	9
2.2	Vývojové prostředí	10
2.3	Dokumentace	10
2.4	Shrnutí.....	11
3	Mikrokontroler	12
4	Vstupně/výstupní piny	15
4.1	Nezbytná teorie.....	15
4.1.1	Frekvence výstupního pinu.....	16
4.1.2	Napěťové úrovně signálů	16
4.2	Praktické ukázky.....	18
4.2.1	Tlačítkem rozsvícená LED	18
4.3	Kvíz.....	28
5	Přerušení.....	29
5.1	Nezbytná teorie	29
5.1.1	Softwarové a hardwarové přerušení	29
5.1.2	Maskování přerušení	30
5.1.3	Příznak přerušení	31
5.1.4	Priorita přerušení	32
5.1.5	Vektor přerušení	33
5.1.6	Obsluha přerušení (ISR).....	34
5.2	Praktické ukázky.....	35
5.2.1	Tlačítkem rozsvícená LED pomocí přerušení	35
5.3	Kvíz.....	41
6	UART.....	42
6.1	Nezbytná teorie.....	42
6.1.1	Protokol komunikace.....	42
6.1.2	UART u STM32	43
6.2	Praktické ukázky.....	45
6.2.1	Příjem a odeslání dat bez knihoven	45
6.2.2	LED rozsvícená povel z počítače	51
6.3	Kvíz.....	54
7	Časovač	55
7.1	Nezbytná teorie	55
7.1.1	Základní režimy časovače	56
7.2	Praktické ukázky.....	59

7.2.1	Blikání pomocí časovače s přerušením	59
7.2.2	Blikání pomocí OC režimu.....	63
7.2.4	Blikání s nastavením délky pomocí IC režimu.....	65
7.2.5	Nastavení jasu LED pomocí PWM režimu	69
7.2.6	Časovač spuštěný čítačem vnějších událostí	71
7.3	Kvíz.....	75
8	ADC	76
8.1	Nezbytná teorie	76
8.1.1	Režimy měření.....	77
8.2	Praktické ukázky.....	79
8.2.1	Přerušované měření dvou kanálů.....	79
8.2.2	Jednotlivé měření dvou kanálů s externím spouštěním.....	82
8.3	Kvíz.....	87
9	DAC	88
9.1	Nezbytná teorie	88
9.2	Praktické ukázky.....	89
9.2.1	Nastavení jasu LED pomocí DAC	89
9.3	Kvíz.....	91
10	SPI.....	92
10.1	Nezbytná teorie	92
10.1.1	Protokol komunikace.....	92
10.1.2	Režimy komunikace	93
10.1.3	SPI u STM32	94
10.2	Praktické ukázky.....	95
10.2.1	Jednosměrná SPI komunikace Master->Slave	95
10.2.2	Jednosměrná SPI komunikace Slave->Master	101
10.2.3	Obousměrná SPI komunikace Master<->Slave	102
10.3	Kvíz.....	105
11	I2C.....	106
11.1	Nezbytná teorie	106
11.1.1	Protokol komunikace.....	107
11.1.2	I2C u STM32.....	109
11.2	Praktické ukázky.....	111
11.2.1	Jednosměrná komunikace Master->Slave a Master<-Slave.....	111
11.3	Kvíz.....	116
12	DMA	117
12.1	Nezbytná teorie	117
12.2	DMA u STM32.....	118
12.3	Praktické ukázky.....	119
12.3.1	DMA přenos z paměti do paměti.....	119

12.3.2	DMA přenos z periferie do paměti	122
12.3.3	DMA přenos z periferie do periferie	126
12.4	Kvíz.....	130
13	Závěr	131
14	Reference.....	133
15	Přílohy.....	134
15.1	Instalace standardních knihoven.....	134

Poděkování

Děkuji firmám

SIEMENS

a



za podporu při psaní této knihy.

1 Úvod

Milý (budoucí) programátore mikrokontrolerů, také už se Ti stalo, že ses chtěl naučit programovat nový typ mikrokontroleru, ale nevěděl jsi, jak začít? Netušil jsi, jaké vývojové platformy a nástroje jsou k dispozici, jaké jsou možnosti mikrokontroleru a jak jej nastavit a používat tak, aby dělal, co od něj chceš?

Pokud ano, jistě jsi využil internetu k hledání nepřehledného množství dokumentů, doporučení, návodů, tipů, příkladů, diskuzí a fór. V tom případě jsi také často zjistil, že velká část z nalezených informací je od uživatelské komunity, tvořena mnohdy domněnkami, fámami, bájemi. A pokud už jsi nějaké řešení našel, většinou neposkytlo odpověď na Tvoji otázku.

Z toho důvodu je zde tato kniha, která Tě provede vším základním, co potřebuješ vědět o programování mikrokontrolerů STM32.

1.1 Motivace knihy

Původním důvodem k napsání této knihy bylo vytvořit studijní podklady pro studenty předmětu „Laboratoře aplikované elektroniky“ (neoficiálně Siemens Embedded Academy) na ČVUT. Tento předmět vedený zaměstnanci firmy Siemens si již několik let dává za cíl naučit studenty, jak programovat mikrokontrolery STM32. Poté, co jsem sepsal kapitoly k jednotlivým přednáškám, mi došlo, že by byla škoda tyto materiály omezit pouze k účelu výuky studentů. Jednotlivé dokumenty jsem spojil v jeden celek – knihu, kterou právě držíte v ruce.

Tato kniha je určena především programátorům mikrokontrolerů, ať už úplným nováčkům či nadšencům. Také těm, kteří již mají nějaké zkušenosti s mikrokontrolery jiných značek (PIC, AVR, MSP430) a nyní by rádi pracovali s mikrokontrolery STM32, ale nevědí, jak začít. Ovšem i profesionálové tuto knihu využijí pro rychlý přechod z jiných kontrolerů na STM32.

Avšak zřejmě největší motivací knihy je ukázat všem, že programování profesionály používaných čipů STM32 není tak obtížné, jak se může po otevření tisícistránkového referenčního manuálu zdát.

Je pravda, že již existují publikace, které o práci s STM32 pojednávají. Tyto knihy ale mají mimo předností i své nedostatky. Prakticky všechny jsou cizojazyčné, což samo o sobě může být například pro středoškolského studenta problémem. Odborné cizojazyčné knihy jsou v ČR špatně dostupné a jejich cena nebývá příznivá. Nahlédneme-li do některých z nich, zjistíme, že mnohdy ani nepokrývají všechny základní periferie kontroleru. Případně se naopak zbytečně detailně zabývají těmi, jež v praxi nejsou příliš často používané. Ukázkové příklady bývají zbytečně složité a matoucí, je k nim potřeba další hardware, nebo využívají neoficiální či zastaralé knihovní funkce.

Všem těmto nedostatkům se snaží tato kniha vyhnout. Postupně dle obtížnosti a logické návaznosti jsou popsány všechny základní periferie v jejich nejpoužívanějších režimech. Příklady jsou velmi jednoduché k pochopení, praktické, psané za použití oficiálních vývojových prostředků a knihoven, jež udržuje výrobce čipů STMicroelectronics. Ke všem příkladům stačí pouze jedna vývojová deska, není tedy nutné, mimo propojovacích drátků, dokupovat či snad sestavovat žádný dodatečný hardware.

1.2 Struktura knihy

Kniha ve svých devíti kapitolách popisuje nejdůležitější periferie, které mají téměř všechny řady mikrokontrolerů STM32.

Každá z těchto kapitol má stejnou strukturu. Po úvodním slovu, k čemu daná periferie obecně slouží, následuje podkapitola s nezbytnou teorií. Ta je určena především těm čtenářům, kteří s kontrolery ještě moc velké

zkušenosti nemají. V této podkapitole se zpravidla nachází blokový diagram dané periferie a jeho popis za účelem snadného pochopení jejího řešení a funkce.

Následující podkapitola pojednává o možnostech, režimech či dalších funkcích dané periferie. Je v ní blíže popsáno, co uživateli může daná periferie nabídnout k tomu, aby snadno dosáhl kýženého výsledku.

Poslední podkapitolou jsou praktické příklady, v nichž je názorně ukázáno, jak periférii nastavit do daného režimu a jak ji používat tak, aby plnila očekávanou funkci.

Na konci každé kapitoly se nachází krátký kvíz, který otestuje, zda si čtenář z dané kapitoly odnesl důležité poznatky. Kvíz je opět určený především pro nováčky.

Každá kapitola předpokládá znalosti z kapitol předchozích, a to především u příkladů. Není tedy doporučeno kapitoly přeskakovat, zvláště pak v případě, kdy čtenář nemá předchozí zkušenosti s žádnými kontrolery.

Zdrojový kód příkladů je zde ve formě obrázků. Důvod je edukativní – čtenář tak nemůže text zkopírovat do vývojového prostředí, ale musí jej skutečně napsat. Tímto způsobem si člověk spíše uvědomí, co vlastně programuje.

Errata a projekty příkladů z této knihy je možné najít na www.programujemekontrolery.cz.

1.3 Fonty textu

Anglické zkratky, názvy nabídek a políček vývojového prostředí a názvy registrů jsou vždy napsány tučnou kurzívou - např. ***AutoReload*** registr. Ač by bylo konzistentnější používat v knize buď pouze české, nebo anglické názvosloví, existuje-li český ekvivalent (např. názvu registru), je použit, jelikož lépe zapadne do věty, která je pak srozumitelnější.

V knize se nachází obrázky nejčastěji v podobě blokových diagramů. V nich jsou některé části zvýrazněny případně i číselně označeny. Na tato označení je v textu odkazováno tučně psaným číslem v závorce – např. **(1)**.

Matematické výrazy či vzorce jsou vždy napsány kurzívou – např. *(počet pulzů + 1)*.

Odkazy na použité zdroje jsou uvedeny číslem v hranatých závorkách – např. [1].

2 Vývojové nástroje

K programování mikrokontrolerů jsou zapotřebí tři základní nástroje.

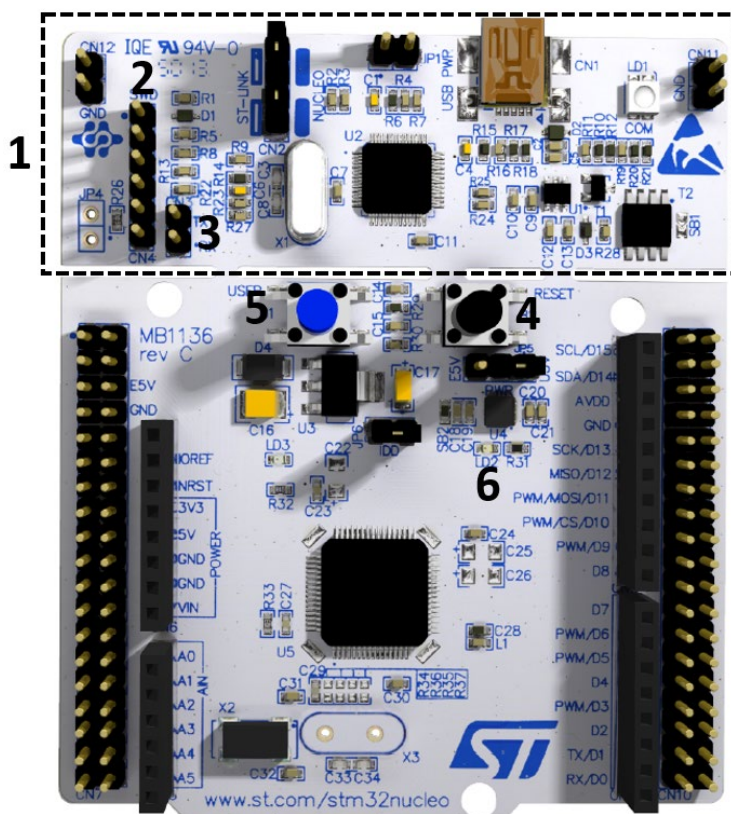
Prvním z nich je samozřejmě samotný mikrokontroler. Nejsnazším způsobem, jak mít k dispozici funkční mikrokontroler připravený k programování a používání jeho periférií, je opatřit si vývojovou desku. Firma STMicroelectronics nabízí širokou škálu cenově příznivých vývojových desek. Mezi nejznámější patří řady desek **Nucleo** a **Discovery**. Ty se od sebe liší tím, že zatímco **Nucleo** obsahuje převážně pouze mikrokontroler a konektory, **Discovery** dost často skýtá také rozličné senzory, tlačítka, LED, displeje a jiné prvky připojené přímo ke kontroleru a připravené k okamžitému použití.

Druhým potřebným nástrojem je programátor mikrokontroleru, který dokáže nahrát binární kód z PC do paměti programu čipu. Tento programátor je možné koupit samostatně, ale výhodou všech vývojových desek firmy STMicroelectronics je, že každá z nich má v sobě programátor a debugger (používaný k ladění programu) již integrovaný.

Zbývá poslední nástroj, ačkoliv jde vlastně o dva, které je však možno opět získat v jednom balíčku. Prvním je vývojové prostředí, ve kterém je možné snadno psát kód a spravovat projekt programu. Druhým je překladač zdrojového kódu, psaného v jazyce C, do strojového binárního kódu, kterému rozumí daný kontroler. Firma STMicroelectronics nabízí zdarma své vývojové prostředí **STM32CubeIDE**, ve kterém je možné napsat zdrojový kód, přeložit jej do strojového a pomocí programátoru jej nahrát do programové paměti mikrokontroleru.

2.1 Vývojová deska

Vývojová deska, která je použita na všechny příklady v této knize, je z řady desek **Nucleo** s názvem **NUCLEO-L073RZ**.



Vývojová deska NUCLEO-L073RZ [1]

Deska obsahuje, mimo mikrokontroleru **STM32L073RZ**, také již zmíněný programátor a debugger sloužící k ladění programu (1). Programátor se nachází v horní části desky, kterou je možné od spodní odlomit. K programování kontrolerů mimo vývojovou desku slouží konektor **CN4** v levé části (2). Horní část také

obsahuje převodník z USB komunikace na **UART** (které je věnována celá kapitola této knihy), jehož vývody **CN3** jsou vpravo od programovacího konektoru (**3**). Ani jeden z těchto konektorů není třeba používat, jelikož programátor i převodník jsou s kontrolerem spojeny můstky mezi horní a dolní částí desky.

Na dolní části desky se nachází dvě tlačítka, jedno z nich - **B2** - slouží k resetování programu mikrokontroleru (**4**), druhé - **B1** - je uživatelské (**5**), které je připojené na pin mikrokontroleru a jež bude často využíván v našich příkladech.

Kromě LED signalizujících funkci programátoru a správné napájení, je na desce umístěna také jedna uživatelská LED **LD2** (**6**), kterou lze pinem mikrokontroleru ovládat.

Na krajích desky pak nelze přehlédnout dva typy konektorů. Vnitřní tvořené dutinkami jsou kompatibilní se všemi rozšiřujícími deskami pro platformu Arduino. Vnější hřebíkové jsou přivedeny ke všem pinům kontroleru.

POZOR!! Ne všechny konektory jsou skutečně připojeny k pinům kontroleru. Někdy je nutné na příslušné místo na desce připájet nulový rezistor, aby došlo k propojení. Důvodem je, že daný pin je již použit například programátorem. Detaily je možné najít v dokumentaci *Nucleo* desky [1].

Kromě výše zmíněného deska ještě nabízí možnost připojení vlastního přesného oscilátoru, místo na měření proudové spotřeby a přepnutí na externí napájení. Nic z toho ale v této knize není využito.

2.2 Vývojové prostředí

Jak již bylo zmíněno, výrobce čipu poskytuje zdarma vývojové prostředí **STM32CubeIDE**, ve kterém jsou vytvořeny všechny příklady této knihy.

Toto prostředí v sobě obsahuje konfigurator periférií čipu, programování a správu projektu, nahrání programu do mikrokontroleru a možnost jeho ladění (debugování).

S instalací vývojového prostředí se automaticky nainstaluje i ovladač programátoru, který je umístěn na *Nucleo* desce, a překladač ze zdrojového kódu na strojový.

STM32CubeIDE je možné po registraci na webových stránkách firmy STMicroelectronics stáhnout zcela zdarma. Nejsnazším způsobem nalezení odkazu ke stažení je zadat do internetového vyhledávače heslo „STM32CubeIDE“ a pravděpodobně hned první odkaz bude mířit na správnou stránku, kde najdete také návod na instalaci a manuál k použití.

Věřím, že není nutné popisovat instalaci vývojového prostředí, která je plně automatická a nainstaluje i všechny nutné ovladače.

2.3 Dokumentace

K práci s mikrokontrolerem a vývojovou deskou přijdou vhod čtyři dokumenty.

Prvním a nejdůležitějším je referenční manuál mikrokontroleru [2]. Tento dokument obsahuje kompletní popis kontroleru, jeho funkcí a periférií. Po jeho otevření je nutné nezaleknout se počtu stránek, jelikož ne všechny jsou stejně důležité. Veškeré kapitoly o perifériích kontroleru, mají stejnou strukturu. Po obecném úvodu a výpisu funkcí periferie následuje blokový diagram periferie, na který bude tato kniha často odkazovat. Kapitola pokračuje popisem jednotlivých funkcí a režimů periferie. I tyto informace jsou často zmíněny v této knize. Poslední částí kapitoly v referenčním manuálu jsou popisy registrů spjatých s danou periférií. Funkce některých z registrů jsou opět popsány v této knize, ale většina z nich může být spravována knihovními funkcemi. Ty jsou použity v příkladech, a proto většinou není důležité se jednotlivými registry zabývat.

Druhým dokumentem je uživatelský manuál vývojové desky *Nucleo* [1]. V něm lze mimo jiné najít, jaké piny kontroleru jsou připojeny k dutinkovým a hřebíkovým konektorům desky či na kterých pinech je připojeno

uživatelské tlačítko a uživatelská LED. Také se v něm nachází informace, které piny nejsou s konektory spojeny vůbec a zda je možné to dodatečně udělat připájením nulového rezistoru na konkrétní pájící plošky na desce.

Dalším důležitým dokumentem je manuál *HAL* knihoven [3] (*Hardware Abstraction Layer*), které jsou použity v příkladech této knihy. Jde o oficiální knihovny vytvořené a udržované firmou STMicroelectronics. V tomto dokumentu se nachází výčet a popis všech knihovnických funkcí k ovládání periférií mikrokontroleru. Jsou zde k nalezení vstupní a výstupní parametry funkcí a jejich význam. Obdobný popis je i v komentářích souborů implementace funkcí. Není tedy nutné mít při práci na programu tento dokument otevřený, stačí mít ve vývojovém prostředí zobrazený daný soubor.

Poslední, spíše doplňkový dokument, je katalogový list (*datasheet*) [4] rodiny mikrokontroleru. V dokumentu jsou uvedena obecná data o mikrokontroleru, jako druhy, počet a stručný popis všech periférií, velikosti pamětí, dostupná pouzdra, statické a dynamické charakteristiky (napájení a spotřeba čipu, teplotní charakteristiky, chyby převodníků...), názvy pinů pouzder, informace o napájení čipu, informace pro návrh plošného spoje a pro osazení. Tento dokument je zde uveden především kvůli důležité tabulce popisující jednotlivé piny kontroleru, která je zmíněna v kapitole o vstupně/výstupních pinech.

Všechny tyto dokumenty přijdou vhod v momentě, kdy se chystáme pracovat s jeho periférií nebo je potřeba k mikrokontroleru připojit externí periférii.

2.4 Shrnutí

Po přečtení této kapitoly by měl mít čtenář k úspěšnému naprogramování příkladů uvedených v této knize připraveny následující nejnutnější věci:

vývojovou desku **NUCLEO-L07RZ**,

USB kabel k připojení desky v počítači,

počítač s nainstalovaným vývojovým prostředím **STM32CubeIDE**,

a **několik drátků** k vzájemnému propojení vnějších konektorů vývojové desky.

S touto nezbytnou výbavou, k níž je doporučeno mít k dispozici zmíněnou dokumentaci, je možné se směle pustit do čtení následujících kapitol, které odhalují programování mikrokontrolerů STM32.

3 Mikrokontroler

Na úvod je vhodné ve stručnosti říci, co je mikrokontroler, z čeho se nutně skládá a jaké může obsahovat periferie.

Mikrokontroler (jinak mikrořadič, jednočipový počítač či mikropočítač) je programovatelná elektronická součástka, jejíž chování je definováno programem uloženým v programové paměti.

Základní části každého mikrokontroleru jsou:

- Mikroprocesorová jednotka (proto je označení mikrokontroleru jako procesor nebo mikroprocesor nepřesné, procesor je totiž pouze součást mikrokontroleru),
- paměť programu,
- operační paměť dat,
- vstupně/výstupní periferie,
- a sběrnice propojující všechny bloky.

Bez těchto částí nemůže kontroler fungovat.

Jako základní část kontroleru by se dal nazvat i zdroj hodinových signálů pohánějící procesorovou jednotku, ale tento zdroj může být i externí (byť dnes již každý kontroler má minimálně jeden zdroj integrovaný).

Všechny tyto základní bloky se svými vlastnostmi liší v závislosti na výrobci a ceně. Kmitočet, na němž pracuje procesor, se může pohybovat od desítek kHz do stovek MHz, paměti programu a dat mohou nabývat velikosti od několika kB do jednotek MB, a vstupně/výstupních pinů může být pouze několik párů, či přes stovku.

Následující tabulka popisuje vlastnosti mikrokontroleru **STM32L073RZ**, se kterým tato kniha pracuje.