

Řízení modelu auta s využitím optického snímače

Lukáš Bajer

Bakalářská práce
2016

 Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
akademický rok: 2015/2016

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Lukáš Bajer**
Osobní číslo: **A12002**
Studijní program: **B3902 Inženýrská informatika**
Studijní obor: **Informační a řídicí technologie**
Forma studia: **prezenční**

Téma práce: **Řízení modelu auta s využitím optického snímače**
Téma anglicky: **The Control of a Car Model Using Optical Sensor**

Zásady pro vypracování:

1. Prostudujte a popište problematiku řízení modelu auta s využitím optického snímače.
2. Sestavte z dodaných komponent model auta řízený mikropočítačem.
3. Navrhněte programové vybavení pro použitý mikropočítač.
4. Navrhněte a realizujte algoritmus pro co nejrychlejší průjezd po předem neznámé dráze s využitím dat z optického snímače.

Rozsah bakalářské práce: -
Rozsah příloh: -
Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

1. BARR, Michael a Anthony J MASSA. Programming embedded systems: with C and GNU development tools. 2nd ed. Sebastopol: O'Reilly, 2006, xxi, 301 s. ISBN 978-0-596-00983-0.
2. GIBSON, J.R. Arm assembly language: an Introduction. Cambridge, UK: ARM Ltd, 2007. ISBN 978-184-7536-969.
3. VALVANO, Jonathan W. Embedded systems: Introduction to the Arm Cortex(TM)-M3 microcontrollers. 2nd ed. s.l.: CreateSpace, 2012, xii, 462 s. ISBN 978-1477508992.
4. SMITH, Warwick A. C programming for embedded microcontrollers. 2nd ed. Susteren: Elektor International Media BV, 2008. ISBN 978-090-5705-804.
5. VÁŇA, Vladimír. ARM pro začátečníky. Praha: BEN - technická literatura, 2009, 195 s. ISBN 978-80-7300-246-6.

Vedoucí bakalářské práce: **Ing. Jan Dolinay, Ph.D.**
Ústav automatizace a řídicí techniky
Datum zadání bakalářské práce: **19. února 2016**
Termín odevzdání bakalářské práce: **27. května 2016**

Ve Zlíně dne 19. února 2016

doc. Mgr. Milan Adámek, Ph.D.
děkan



prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu

Prohlašuji, že

- beru na vědomí, že odevzdáním diplomové/bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová/bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji diplomovou/bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou/bakalářskou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování diplomové/bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové/bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové/bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na diplomové/bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne 23. 5. 2016



.....
podpis diplomanta

ABSTRAKT

Cílem práce je navrhnout a vytvořit programové vybavení modelu samořídícího vozidla. Vozidlo má za úkol projet neznámou dráhu, která má na okrajích černé čáry. Tyto čáry slouží pro vedení modelu auta po dráze. Pro snímání čar slouží řádková kamera a na základě dat z ní je ovládáno řízení modelu auta. Základem je model auta a jeho komponenty určené pro soutěž The NXP Cup. Jako vývojové prostředí pro programové vybavení modelu auta je zvolen CodeWarrior a je použitý programovací jazyk C.

Klíčová slova: Samořídící vozidlo, řádková kamera, The NXP Cup

ABSTRACT

Purpose of this thesis is to design and create software for self driving car model. The vehicle is designed to pass through nun known pathway, which has black lines at the edges. These lines are used to guide the car model on the runway. For scanning the black lines a line scan camera is used. Based on the data from this camera the model car is controlled. The basis of the model of the car and its parts designed for competition The NXP Cup. For the software of the model car CodeWarrior is selected and C programming langure is used.

Keywords:Self driving car, line scan camera, The NXP Cup

Chtěl bych poděkovat vedoucímu práce panu Ing. Janu Dolinayovi, Ph.D. za připomínky a odborné rady k mé práci, které mi byly přínosem.

Prohlašuji, že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

OBSAH

ÚVOD	9
I TEORETICKÁ ČÁST	10
1 SAMOŘÍDICÍ VOZIDLA	11
1.1 SENZORY	11
1.1.1 Radary	12
1.1.2 Lidary	12
1.1.3 GPS	12
1.1.4 Kamery	12
1.1.5 Ultrazvukové snímače	13
1.2 SOUČASNÁ AUTONOMNÍ VOZIDLA	13
1.2.1 Google samořídící vozidlo	13
1.2.2 Tesla Model S	14
2 MODELY SAMOŘÍDICÍCH VOZIDEL A ROBOTI	15
2.1 VÝVOJOVÉ KITY	15
2.1.1 Arduino	15
2.1.1.1 Vývojový kit ARDUINO LEONARDO	16
2.1.2 Vývojové desky společnosti NXP	17
2.2 ARCHITEKTURA PROCESORŮ ARM	17
2.2.1 Historie	17
2.2.2 Přehled procesorů	18
2.2.2.1 ARM Cortex-M0+	18
2.3 ROZŠÍŘUJÍCÍ MODULY	19
2.3.1 Motory	19
2.3.2 Serva	19
2.3.3 Moduly s kamerami	20
2.3.4 Rozšiřující modul se zobrazovačem LCD	20
2.3.5 FRDM-FXS-MULT2-B	20
2.3.6 FRDM-17510EJ-EVB	21
3 SOUTĚŽE ROBOTŮ A SAMOŘÍDICÍCH VOZIDEL	22
3.1 FORMULA-E ROBORACE	22
3.2 ROBOORIENTEERING	22
3.3 EUROBOT	22
3.4 ROBOT CHALLENGE	22
3.5 THE NXP (FREESCALE) CUP	22
II PRAKTICKÁ ČÁST	24
4 POPIS HW VYBAVENÍ MODELU AUTA	25

4.1	PODVOZEK	25
4.2	ŘÍZENÍ (SERVO)FUTABA S3010	25
4.3	ZADNÍ POHON	26
4.4	ŘÁDKOVÁ KAMERA S DRŽÁKEM	26
4.5	FRDM-TFC	27
4.6	MIKROPOČÍTAČFRDM-KL25Z.....	28
4.7	BATERIE	29
5	POPIS SW VYBAVENÍ MODELU AUTA	30
5.1	POUŽITÉ KNIHOVNY	30
5.1.1	Funkce pro čtení dat z kamery	30
5.1.2	Funkce pro ovládání servomotoru.....	31
5.1.3	Funkce pro ovládání zadního pohonu	31
5.2	VLASTNÍ ŘEŠENÍ ALGORITMU	31
6	ALGORITMUS PRO PRŮJEZD	33
	ZÁVĚR	34
	SEZNAM POUŽITÉ LITERATURY.....	35
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....	37
	SEZNAM OBRÁZKŮ	38
	SEZNAM PŘÍLOH.....	39

ÚVOD

Kolem nás se objevuje čím dál více autonomních vozidel či robotických systémů. Brzy autonomní vozidla budou našim standardem a velkým přínosem budou v logistice, kde nebudou potřeba řidiči nákladních vozidel a sníží se tak nehodovost na pozemních komunikacích. V této práci se zabývám samořídícími vozidly a také jejich modely, což je spojeno i s programovatelnými procesory. Praktická část je založena na naprogramování modelu samořídícího vozidla řízeného pomocí optického snímače. Tento model má za úkol projet neznámou dráhou, jehož jedinými vodítky budou černé čáry na okraji dráhy. K detekci těchto čar slouží řádková kamera, která vrací obraz jeden řádek o velikosti 128 bitů. Na základě vyhodnocených dat z kamery je ovládáno řízení modelu auta, které je realizováno servomotorem.

Programové vybavení procesoru modelu auta je napsáno v jazyce C a jako vývojové prostředí je použit CodeWarrior, také byly využity knihovny od firmy NXP pro rozšiřující desku FRDM-TFC.

V samotné teoretické části jsou popsány samořídící vozidla současnosti. Tématem práce je model samořídícího vozidla a jeho základem je CPU, které se stará o funkčnost samotného modelu auta. Proto je popsáno několik vývojových kitů osazených různými procesory. Další částí modelu auta je rozšiřující modul a tak jsou uvedeny i rozšiřující moduly a komponenty propojitelné s vývojovými kity.

V praktické části je popsán celý model auta a jeho komponenty. Také je popsáno programové vybavení a algoritmus použitý při průjezdu dráhou i s vývojovým diagramem. Práce slouží i jako tutoriál a proto bylo uvedeno zapojení všech komponent k vývojovému kitu.

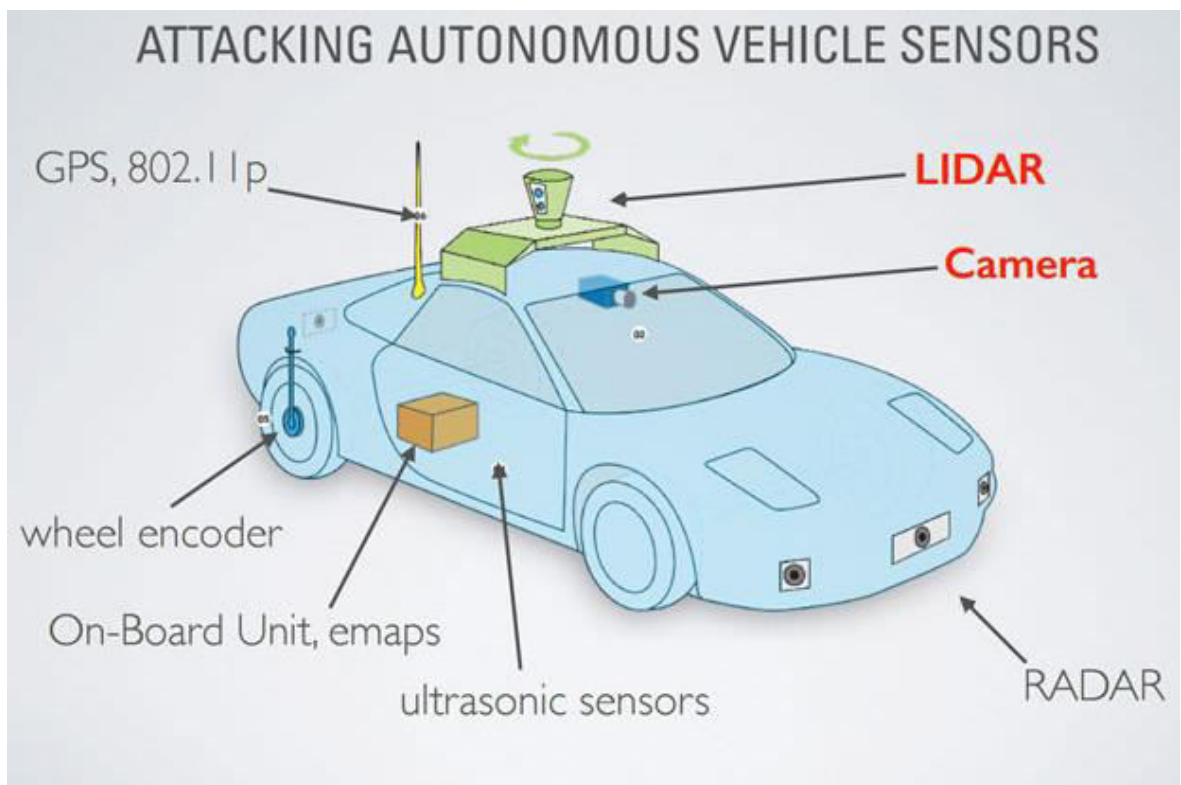
I. TEORETICKÁ ČÁST

1 SAMOŘÍDICÍ VOZIDLA

Samořídící vozidla, také nazývané autonomní, jsou vozidla fungující v silničním provozu bez zásahu lidského faktoru. Využívají různých senzorů a snímačů ke snímání okolí aby nedošlo ke kolizím. [1]

1.1 Senzory

Mezi používané snímače patří radary, lidar, GPS, kamery a ultrazvukové senzory. Data ze všech těchto senzorů jsou přivedena do centrálního počítače, který je vyhodnocuje a podle potřeby ovládne samotné vozidlo. Vozidla také mezi sebou dokáží komunikovat a tak se dokáží navzájem upozornit například na nebezpečí vzniklé na vozovce.



Obrázek 1 Senzory používané u autonomních vozidel [2]

1.1.1 Radary

Samořídící vozidla využívají radary pro snímání jedoucích vozidel před i za. Jsou umístěny v předním i zadním nárazníku. Už i v dnešní době v sériových vozech jsou využívány pro adaptivní tempomaty. Slouží k detekci jedoucího vozidla před řízeným vozidlem a v závislosti na vzdálenosti buď automaticky vozidlo zrychluje či zpomaluje, je tak vyloučena možnost střetu s vozidlem jedoucím před řízeným vozidlem.

Jejich princip v jednoduchosti spočívá ve vyslání elektromagnetického záření o určitém výkonu a frekvenci. Při kolizi vyslaného paprsku s objektem dojde k odrazu a signál je zpět přijímán. Výpočtem časového rozdílu mezi odesláním a přijetím paprsku se vypočítá vzdálenost od objektu.

1.1.2 Lidary

Jedná se o jeden z nejdůležitějších senzorů samořídícího vozidla. Vytváří 3D mapování okolí vozidla a dokáže tak zamezit střetu s jakýmkoliv překážkami. Skládají se z několika laserů v řadě a otáčejí se stále dokola pro zmapování celého okolí vozidla.

Jejich princip je podobný jako u radarů, ale zde je využíván laserový paprsek. Vyšle se laserový paprsek, v případě kolize s objektem dojde k odrazu a laserový paprsek je přijat a opět je vypočítána vzdálenost pomocí časového rozdílu mezi vysláním a přijmutím paprsku.

1.1.3 GPS

GPS lokátory jsou umístovány do vozů už nějakou dobu pro zjištění polohy vozidla, například ve firmách. U samořídících vozidel mají stejnou funkci, dokáží zjistit aktuální polohu vozidla a dokáží tak plánovat nejvhodnější trasu k cíli. Určení polohy má odchylku i několik metrů a tak se porovnávají údaje s dříve získanými záznamy o poloze.

1.1.4 Kamery

Opět i tyto senzory jsou používány i v současných vozidlech, která se tak téměř stávají samořídícími. Hlídkají vyjetí z jízdního pruhu, současná vozidla buď upozorní, že řidič vyjíždí z jízdního pruhu, pokud nepoužil směrový ukazatel nebo dokonce dokáží samy ovládnout řízení a vrátit vozidlo zpět do jízdního pruhu. V samořídících vozidlech slouží k udržení vozidla v jízdním pruhu. Pomocí kamer je samozřejmě i možné monitorovat

dopravní značení, opět je tato funkce v dnešních vozidlech a dokáže upozornit řidiče například na sníženou maximální povolenou rychlost.

1.1.5 Ultrazvukové snímače

Používají se dlouhou dobu v dnešních vozidlech a dokonce jsou téměř standardem výbavy. Používají se v tzv. parkovacích asistentech pro upozornění na překážky za vozidlem. Jsou umístěny v zadním nárazníku někdy i v předním. V závislosti na vzdálenosti od předmětu informuje řidiče o nebezpečí střetnutí. V samořídících vozidlech, zejména u vozidla google, je umístěn na zadním kole a získává informaci o pohybu vozidla a také informaci o výskytu cizích předmětů u zadní části vozidla.

Jejich princip spočívá ve vyslání akustického signálu a v závislosti na časové odchylce mezi vyslaným a přijatým signálem vypočítá vzdálenost předmětu.

1.2 Současná autonomní vozidla

V dnešní době je čím dál více autonomních vozidel. Už i sériové vozy jsou téměř samořídící, mají různé asistenty brzdění, adaptivní tempomaty, hlídání jízdního pruhu a jiné. Některé vozidla dokáží i samy zaparkovat. První pokusy samořídících vozidel probíhaly už od roku 1950. Podle studie BI intelligence bude v roce 2020 na cestách jezdit 10 milionů samořídících vozidel. [3]

1.2.1 Google samořídící vozidlo

Společnost Google je v této oblasti díky svému samořídícímu vozidlu velice medializovaná. První vozidlo, které představili tak, aby bylo schopné samotného provozu byla Toyota Prius. Další vozy, které dostala pro svůj vývoj byly Lexus RX450h. Google dokonce sestrojil vlastní vozidlo vybavené všemi výše zmiňovanými senzory včetně lidarů s 64 laserovými paprsky. Google své vozy testuje v silničním provozu už od roku 2012 a dopomohla jim k tomu změna zákona v Nevadě, která dovoluje testování samořídících vozidel v tomto státě.

Google samozřejmě prošel několik kritik a nedopomohlo tomu i několik nehod. Většina těchto nehod je zapříčiněna vlivem lidského faktoru. [4]



Obrázek 2 Google samořídící vozidlo [5]

1.2.2 Tesla Model S

Vozidla značky Tesla nejsou vedeny jako samořídící vozidla, sice dokáží převzít řízení, ale za stálé pozornosti řidiče, mají tzv. autopilota. Elon Musk, šéf Tesly, chce místo lidarů použít více ultrazvukových senzorů, které by snímaly celé okolí vozidla, také radary a kamery. S novým update software je tesla sama schopna měnit i jízdní pruh, najít parkovací místo a paralelně zaparkovat. Mezi standardní vlastnosti jsou hlídání vzdálenosti od protijedoucího vozidla, tzv. adaptivní tempomat, jízda v jízdním pruhu apod. [6]

2 MODELY SAMOŘÍDÍCÍCH VOZIDEL A ROBOTI

V dnešní době si postavit elektronické zapojení není nic složitého a sestrojít si a naprogramovat vlastního robota taktéž. Návodů na internetu je spousta a dostupných technologií k sestavení je také dostatek. Modely těchto vozidel či robotů jsou řízeny mikrokontroléry a pro jednodušší aplikaci v modelu, či seznámení s jejich programováním slouží tzv. vývojové kity.

2.1 Vývojové kity

V dnešní době mnoho výrobců prodává vývojové kity a různé moduly pro zaučení programování. Cena těchto kitů se pohybuje i do 1000 Kč a poskytují mnoho zajímavých způsobů zapojení. Nejznámější výrobci jsou NXP(freescale), Arduino, Microchip a STM. Samotné Arduino je už tak zjednodušené svými knihovnamí, že sestrojít si nějaký lehký obvod s jednoduchým programovým vybavením dokáže téměř každý.

2.1.1 Arduino

Arduino je open-source platforma jak HW tak i SW. Vývojové kity této značky stejně jako ostatní jsou schopny číst vstupy například ze světelných senzorů, tlačítek apod. A na základě získaných informací rozsvítit LED diodu, roztočit motorek či poslat paket přes ethernet. Mikroprocesor na desce se programuje pomocí arduino software v jazyce Wiring, který je odvozený od jazyka C. Komunikuje s PC přes USB port.

Původně bylo arduino navrženo v institutu Ivrea pro studenty pro rychlé a jednoduché prototypování. Postupem času se arduino rozvíjelo od 8bitových kitů po desky komunikující přes internet. Dnes mají tyto desky zastoupení v mnoha vestavěných systémech jako jsou například i 3D tiskárny. Díky tomu, že SW i HW jsou open-source, existuje dost příspěvků od uživatelů arduina a velice zajímavých nápadů k sestavení například svého robota.

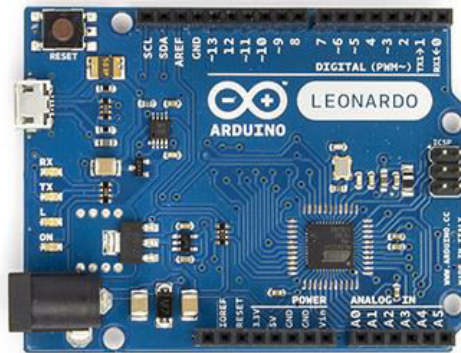
Mezi výhody arduina patří samozřejmě, že jsou open-source. Výhodou je i jejich jednoduchost a díky tomu jsou uživatelé schopni použít arduino do každého zapojení.[7]



Obrázek 3 Seznam produktů Arduino [8]

2.1.1.1 Vývojový kit ARDUINO LEONARDO

Jako příklad vývojových kitů od společnosti Arduino jsem si vybral kit Arduino Leonardo. Tento kit obsahuje procesor ATMEGA32U4, jedná se o 8bit AVR CPU. Taktovací frekvence je 16MHz, paměť 32kB typu flash a 2,5kB paměť typu SRAM. Počet programovacích vstupů/výstupů, který tento procesor poskytuje je 26. Pro komunikaci můžeme použít 1x USB, 2x SPI, 1x I2C, 1x UART. Obsahuje i analogové vstupy a disponuje 10bitovým AD převodníkem. Samotná deska neobsahuje mnoho periférií, ale je spousta variant například s ethernetovým portem a samozřejmě je k těmto deskám spousta doplňujících modulů. V porovnání s deskou FRDM-KL25Z, která je popsána v následující kapitole, není Arduino Leonardo dobře hardwarově vybaveno, ale díky své jednoduchosti, je arduino oblíbené mezi fanoušky „bastlení“. Cena činí kolem 600kč, ale existují i verze mikro, ale neobsahují programátor a tak je potřeba propojit s nějakým programátorem či vývojovým kitem, který programátor má implementovaný na desce. Další možností jsou čínské kopie, které jsou levnější a jejich funkčnost není takovým rozdílem oproti originálu.[8]



Obrázek 4 Vývojový kit Arduino Leonardo [8]

2.1.2 Vývojové desky společnosti NXP

Vývojové desky freedom od společnosti NXP se vyznačují svou malou velikostí a nízkou spotřebou. Tyto desky se dají připojit jako mass-storage pro rychlé nahrání programu v binární podobě přes virtuální sériový port nebo přes klasický programový port. Desky jsou osazeny procesory pod názvem kinetis, které jsou založeny na technologii firmy ARM, jejich řady nesou označení E, L, K, KEA, V, W.

Podobně jako arduino i desky této značky jsou jednoduché pro začátečníky. Je zaručena jejich kompatibilita právě s deskami arduino R3. Jsou také kompatibilní i s jinými deskami výrobců třetích stran.[9]

2.2 Architektura procesorů ARM

Procesory s architekturou ARM se vyznačují svou nízkou spotřebou a také nemají žádné nároky na chlazení. Díky těmto výhodám se používají v mobilních zařízeních, které jsou napájeny z baterií. Mnoho výrobců vyrábí procesory s touto architekturou jako jsou Texas Instruments, Atmel, NXP, STMicroelectronics, Samsung apod.[10]

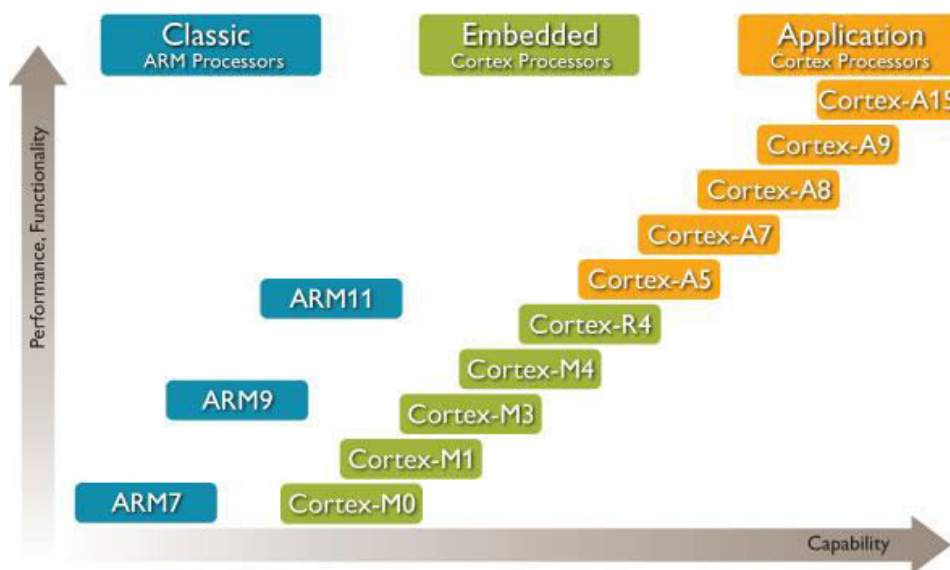
2.2.1 Historie

Dříve se procesory s touto architekturou označovaly jako *Advanced RISC Machine*, které vyvíjela firma *ARM Holdings*. Využívají omezenou instrukční sadu RISC, díky níž jsou procesory jednoduché a rychlé. Dále využívají 32bitovou šířku slova. Firma *ARM Holdings* první procesor vyvinula v roce 1984. Později se přestala věnovat výrobou procesorů a byla pouze duševním vlastníkem technologií a jejich vývoje. Poskytovala

licence k výrobě procesorů architektury ARM jiným společností. V roce 2007 byla ARM architektura procesorů obsažena v 98% každoročně prodaných mobilních telefonů.

2.2.2 Přehled procesorů

Z počátku byla jádra označována ARM1, ARM2, ARM3. Přišel nástup ARM6, například procesor ARM610 pro PDA firmy Apple, která získala majetkovou účast. V roce 1993 ARM7 a později se zlepšením pod označením ARM7 TDMI a stal se nejlépe prodáváním procesorem ARM. Přibyla i jádra s označením ARM8, ARM9 TDMI, ARM10, ARM11 a od architektury verze ARMv7 začali přejmenovávat jádra na Cortex. Jádra Cortex jsou používány v mobilních zařízeních pod označením Cortex-A5, Cortex-A8, Cortex-A9. V oblasti mikrokontrolérů pro vestavěné systémy se označují Cortex-M0, Cortex-M0+, Cortex-M3, Cortex-M4, Cortex-M7. Další označení Cortex-R4, Cortex-R5, Cortex-R7.[11]



Obrázek 5 Přehled procesorů architektury ARM [12]

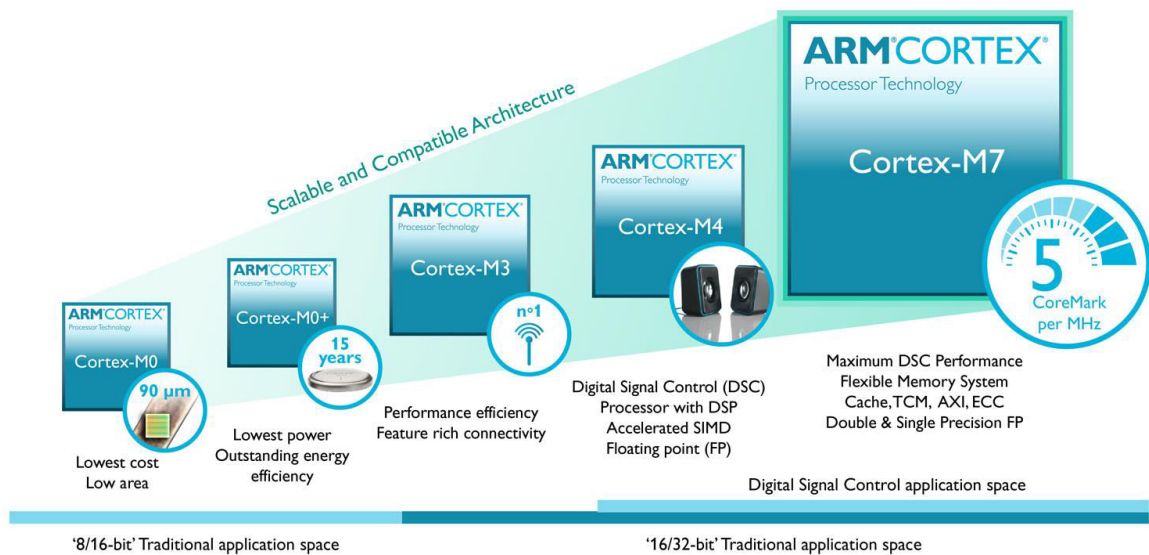
2.2.2.1 ARM Cortex-M0+

Procesor MKL25Z128VLK4 použitý v této práci je architektury ARM Cortex-M0+. Jedná se o levné a malé 32bitové procesory. Cortex-M0+ má kompletní instrukční sadu kompatibilní s Cortex-M0. Využívají architekturu ARMv6-M.[13]

Procesory využívající jádro Cortex-M0+:

- Atmel SAMD, SAMR, SAML, SAMC
- Freescale (NXP) Kinetis E, EA, L, M, V1, W0
- NXP LPC800, LPC11E6x, LPC11U6x

- STMicroelectronics STM32 L0



Obrázek 6 Procesory s jádrem Cortex-M [14]

2.3 Rozšiřující moduly

Pro dosažení nějaké funkčnosti, či sestavení samotného robota či modelu auta je zapotřebí k vývojovému kitu něco připojit. Jak už bylo zmíněno tak těchto rozšiřujících modulů je dostatek a kompatibilní jsou téměř se všemi výrobci, například je kompatibilita mezi deskami arduinoafreedom deskami od NXP. Nemusí se však jednat přímo o nějaké sofistikované desky (moduly) jako například deska s LCD displejem, ale může se jednat i o jednotlivé součástky, jako jsou serva, motory, LED diody, snímače, tlačítka, klávesnice, kamery apod. V následujících kapitolách budou popsány periferie, které jsou použity v této práci a také pro ukázkou nějaké rozšiřující moduly.

2.3.1 Motory

Motory dokáží převést elektrickou sílu na mechanický pohyb. U modelů mají velké využití pro pohyb aut vpřed, pohyb ramen u robotů, otáčení vrtulí u dronů a jiné. Mohou být využity DC motory, krokové motory nebo serva. Jejich ovládání je provedeno pomocí různých driverů většinou v zapojení do H-můstku.

2.3.2 Serva

Oproti klasickým motorům je rozdíl v tom, že u serva dokážeme nastavit přesnou polohu a otáčky serva jsou o dost menší. Využívají se u modelů pro přesný pohyb například pro natočení kol u modelů aut. Servo se skládá z DC motorku a převodovky,

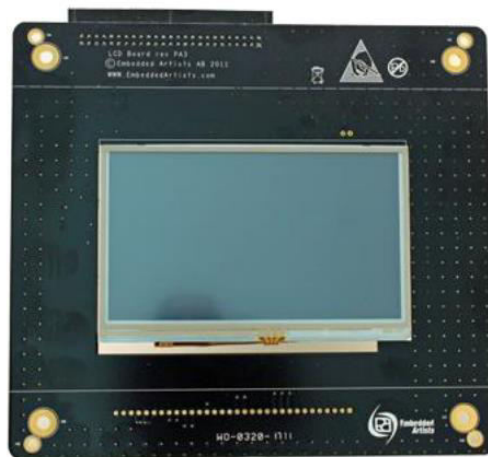
kteřá umožňuje servu dosáhnout velkou sílu. Ze serva vedou 3 drátky, GND, napájecí (většinou +5V) a řídicí, kde je z CPU poslán PWM signál s danou šířkou pulzu pro otočení serva o daný úhel.

2.3.3 Moduly s kamerami

Stejně jako u samořídících vozidel, tak i u modelů je potřeba sledovat okolí, aby se předešlo kolizím. Právě pro takové snímání mohou být použity kamery. Někdy jsou používány pouze pro možnost vzdáleného ovládání (zpětná vazba o pozici) nebo pro pořizování snímků.

2.3.4 Rozšiřující modul se zobrazovačem LCD

Následující rozšiřující moduly včetně tohoto jsou zde uvedeny jako příklad možných periférií, které se mohou připojit k vývojovým kitům. Rozšiřující modul je od společnosti NXP. Jeho rozlišení je 800x480, velikost displeje je 7“ a použitý typ zobrazovače je TFT. Zajímavostí je, že se jedná o dotykový resistivní displej. Komunikace probíhá přes I2C rozhraní. Připojení ke kitu je pomocí 50ti pinového konektoru. Cena se pohybuje od 3000kč.[15]

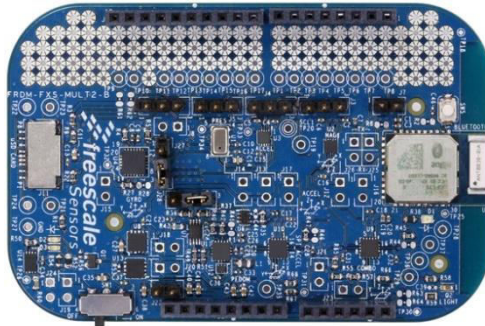


Obrázek 7 Rozšiřující modul s LCD displejem [15]

2.3.5 FRDM-FXS-MULT2-B

Jedná se o desku pro NXP sensory s bluetooth. Mezi podporované sensory patří 3-osý 14bitový akcelerometr *FXLS8471Q*, 3-osý 12bitový akcelerometr *MMA8652FC*, digitální sensor – 3-osý akcelerometr + magnetometr *FXOS8700CQ*, 3-osý magnetometr *MAG3110*, absolutní snímač tlaku v rozsahu 20kPa až 110kPa *MPL3115A2*, 3-osý digitální gyroskop

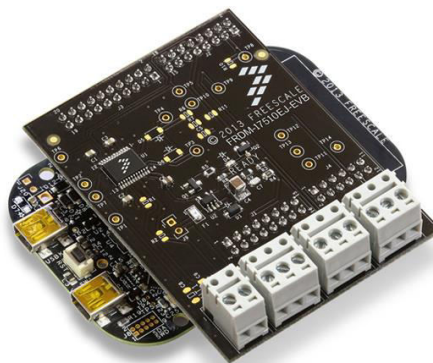
FXAS21002C, krokomeř MMA9553L. Deska nabízí snímání ve 12 osách a komunikaci přes bluetooth.[16]



Obrázek 8 FRDM-FXS-MULT2-B od NXP [16]

2.3.6 FRDM-17510EJ-EVB

Tato deska je osazena MPC17510 monolitickým H-můstkovým driverem pro ovládání malých DC motorků. Deska je navržena pro desku FRDM-KL25Z, ale je kompatibilní s dalšími deskami FRDM a arduinem. Silová část je oddělena od procesoru a chráněna pojistkou.[17]



Obrázek 9 FRDM-17510EJ-EVB od NXP [17]

3 SOUTĚŽE ROBOTŮ A SAMOŘÍDÍCÍCH VOZIDEL

Pro podporu a nalákání co nejvíce mladých nadšenců v oblasti elektroniky a IT se pořádají různé soutěže v této oblasti robotiky. Nejvíce úspěšní jsou v tomto v Japonsku. Robotických soutěží se po světě pořádá několik, v následujících podkapitolách jsou uvedeny některé z nich.

3.1 Formula-E Roborace

Jedná se o soutěže elektrických autonomních formulí. Všechny vozidla budou stejná a tak bude záležet na optimálním algoritmu pro průjezd dráhou a nejrychlejší čas. Závod má trvat 1 hodinu. První závody se mají uskutečnit v roce 2016-2017.[18]

3.2 RoboOrienteering

Tato soutěž se koná u nás v České republice v Rychnově nad Kněžnou. První kolo se konalo v roce 2010. Úkolem soutěže je dopravit náklad do cíle. Těsně před startem vozidla/roboty jsou předány souřadnice cíle a vozidlo/robot si musí najít neoptimálnější trasu a samozřejmě se vyhnout překážkám. Hodnotí se dopravení nákladu v co nejrychlejší čas.[18]

3.3 Eurobot

Soutěž Eurobot je původem z Francie a zúčastňují se jí autonomní roboti. Soutěž byla založena v roce 1994. Jedná se o soupeření dvou robotů na hrací desce, kde jsou různé disciplíny a vždy záleží na splnění úkolů a nejrychlejší čas. Soutěžní kola probíhají po celé Evropě a bývají i v České republice v Praze.[18]

3.4 Robot Challenge

Rakouská soutěž kde se opět utkávají autonomní roboti. V soutěži je několik kategorií jako například průjezd dráhou (jízda po čáře), sběr předmětů a různé závody. Tato soutěž je určena pro všechny kdo si doma vyvíjí různé autonomní roboty a umožňuje jim je prezentovat.[18]

3.5 The NXP (freescale) Cup

Soutěž pořádá, jak název napovídá, společnost NXP. Jedná se o závody modelů autonomních aut, kde je cílem projet dráhu do cíle. Jediným vodičkem pro auto je černá

čára či čáry na okrajích dráhy a pomocí řádkové kamery se musí auto udržet na dráze. V cíli je značení, na kterém auto musí zastavit případně s odchylkou 1 metr. Záleží na aktuálních pravidlech soutěže. Dráha má několik nástrah jako jsou kopce, zatáčky, křižovatky, cesta s vlněnými čárami a někdy i tunely.[19]

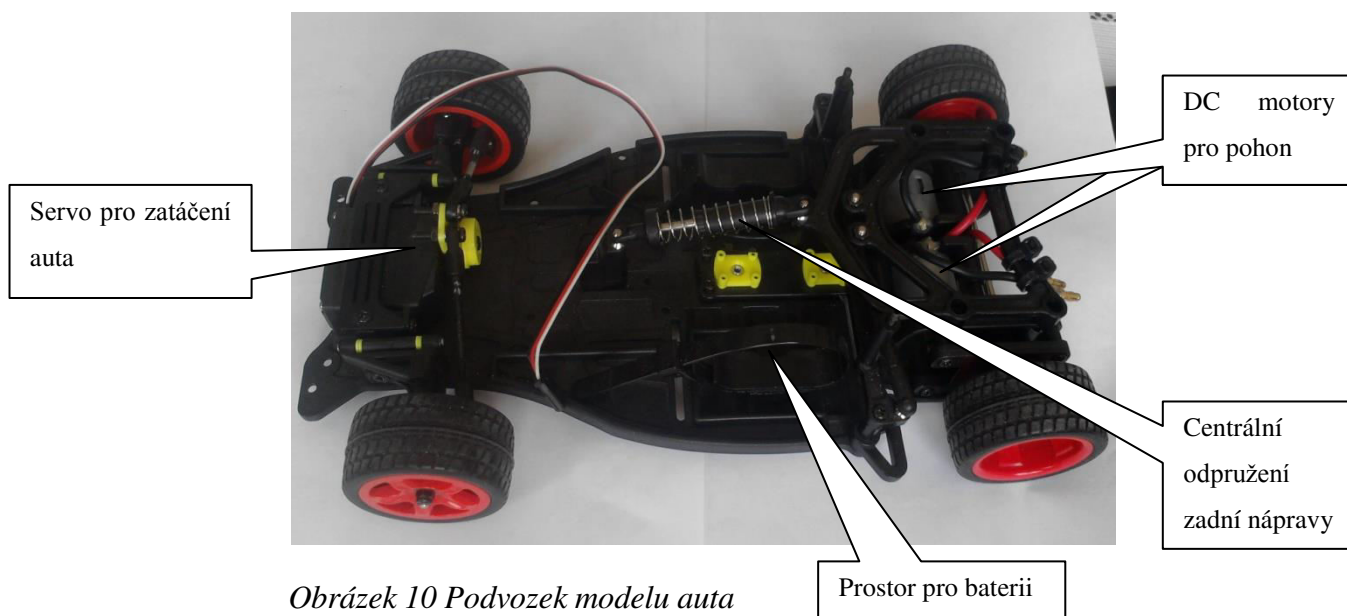
II. PRAKTICKÁ ČÁST

4 POPIS HW VYBAVENÍ MODELU AUTA

Model auta použitý v této práci se používá právě v soutěžích The NXP Cup. Sestaveno je z následujících komponent.

4.1 Podvozek

Podvozek modelu auta je klasická konstrukce, kterou používají modeláři rádiem řízených modelů. Zadní náprava je centrálně odpružena, zadní kola jsou oddělena a každé kolo je poháněno jedním DC motorem. Přední kola jsou také odpružena a o zatáčení se stará servo.



4.2 Řízení (servo)Futaba S3010

Zatáčení modelu auta je realizováno pomocí servo motoru označeného Futaba S3010. Vstupní napětí je v rozsahu 4,8 – 6,0V. Točivý moment při 4,8V je 5,18kg/cm a při 6,0V je točivý moment 6,48kg/m. Rychlost otáčení při 4,8V je 60° za 0,2s a při napětí 6,0V jsou otáčky 60° za 0,16s. Převodovka tohoto serva je plastová a motor je 3-pólový. Celé servo váží 41g.[20]



Obrázek 11 Použité servo v modelu autu [20]

4.3 Zadní pohon

O pohon modelu auta se starají 2 DC motory, které jsou umístěny každý na jednom kole. Řídicí napětí těchto motorů je 7,2V. Točivý moment se pohybuje kolem 80g/cm. Otáčky bez zátěže 16000 +/- 3200 otáček za minutu a proud bez zátěže maximálně 220mA. Pracovní teplota je v rozsahu 10 až 30°C. Otáčky motoru jsou z převodovány přes ozubená kolečka a točivý moment je přiveden na zadní kola.[21]

4.4 Řádková kamera s držákem

Pro zpětnou vazbu o poloze modelu auta se stará rozšiřující deska s kamerou osazená čipem TSL1401CL. Kamera snímá obraz v rozlišení 1x128px, tudíž se jedná o záběr pouze jednoho řádku. Úkolem kamery je dokonale rozeznávat černou a bílou a proto je uvnitř vestavěný zesilovač pro lepší rozeznání černé a bílé. Kamera má možnost zaostření pomocí změny vzdálenosti čočky díky závitu. Deska má 5 pinů, které jsou pro připojení GND, napájecí napětí 5V, CLK vstup pro hodiny pro synchronizaci přijatých bitů, SI vstup pro příkaz zaslání zprávy, AO analogový výstup s pořízenými daty.[22]



Obrázek 12 Použitá kamera

Držák kamery slouží pro umístění kamery ve vhodné vzdálenosti, aby došlo k dokonalému snímání dráhy před autem. Držák byl navržen a vyroben v rámci této práce. Byl vytisknut na 3D tiskárně z materiálu označovaného jako PLA. Konstrukce byla uzpůsobena pro případnou kalibraci úhlu natočení kamery pro dokonalejší záběr dráhy před autem.



Obrázek 13 Držák kamery

4.5 FRDM-TFC

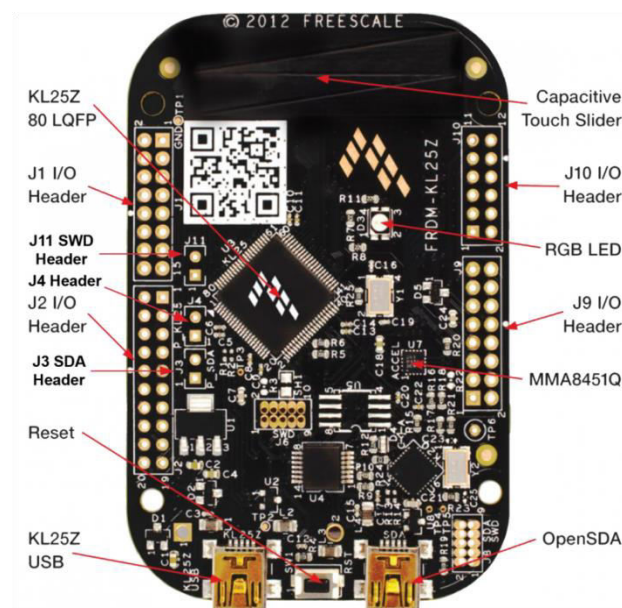
Tuto desku opět využívám ve své práci a je kompatibilní pouze s vývojovými kity řady kinetis od společnosti NXP (freescale). Na desce se nachází 2x driver DC motorů s označením MC33887APVW, jedná se o provedení H-můstek a výstupní napětí může být v rozsahu 5 až 28V. Můžeme zde najít i výstup pro servo, vstup pro řádkovou kameru a vstup pro enkodér. Pro manuální ovládání slouží 2 tlačítka, 2 potenciometry, 4 DIP přepínače a pro zobrazení můžeme použít 4 LED diody určené pro zobrazení stavu baterie, externího napájení. Tato deska je vytvořena pro soutěže freescale cup, které pořádal freescale nyní NXP.[23]



Obrázek 14 Rozšiřující deska FRDM_TFC

4.6 MikropočítačFRDM-KL25Z

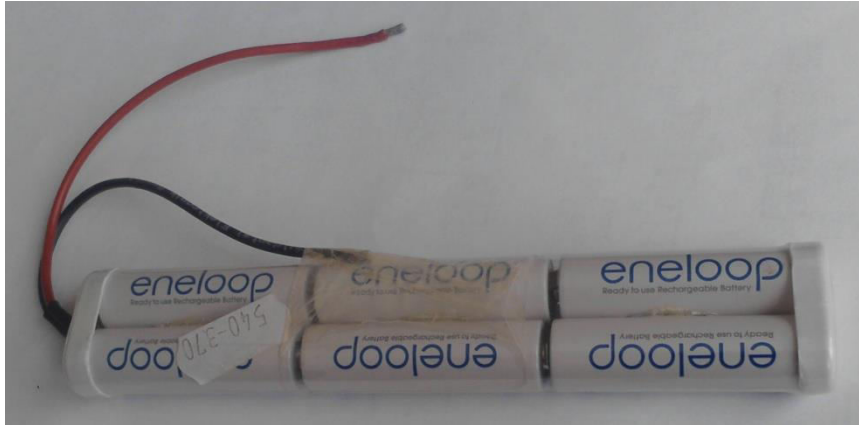
Tento vývojový kit byl použit v bakalářské práci pro řízení modelu auta. Deska kitu je osazena mikrokontrolérem MKL25Z128VLK4 architektury ARM Cortex M0+ , který disponuje taktovací frekvencí čipu 48MHz, obsahuje i paměťové uložení typu flash o velikosti 128kB a také paměť pro ukládání dat o velikosti 16kB typu SRAM. Umožňuje také připojení přes několik sériových rozhraní, jako jsou 1x USB, 2x SPI, 2x I2C, 3x UART. Procesor obsahuje 66 vstupů/výstupů z nichž některé jsou obsazeny moduly na desce. Nesmí také chybět PWM výstupy, 16bit AD převodník a 12bit DA převodník. Co se týče samotné desky, tak na ní můžeme najít 3-osý akcelerometr, RGB LED diodu a dotykovou kapacitní plošku. Cena tohoto kitu se pohybuje kolem 370kč.[24]



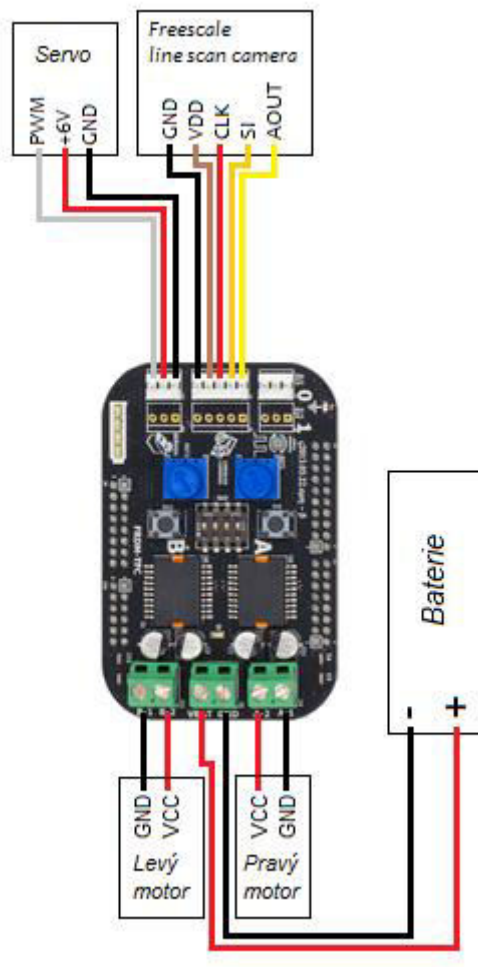
Obrázek 15 Vývojový kit FRDM-KL25Z [24]

4.7 Baterie

Model auta je napájen z 6ti článkové baterie. Každý článek je realizován nabíjecí tužkovou baterií NiMH o napětí 1,2V a kapacitě 1900mAh. Baterie označení Eneloop HR-3UTGA a vyznačují se svou dlouhou životností, která činí až 1800cyklů nabíjení.



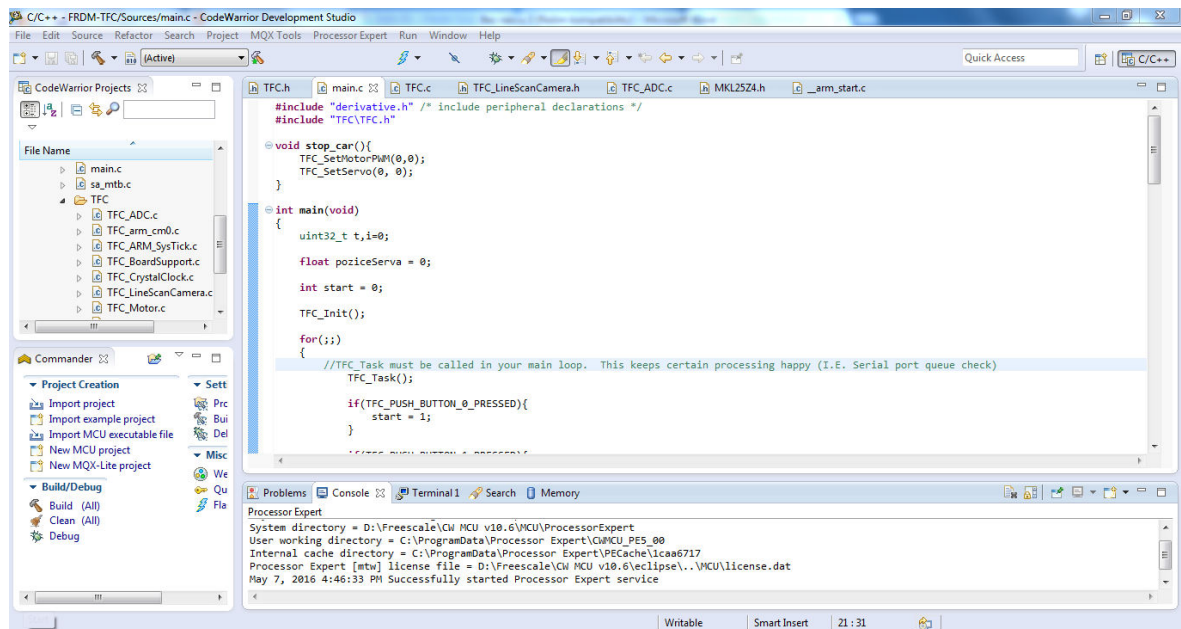
Obrázek 16 Baterie pro napájení modelu



Obrázek 17 Blokové schéma zapojení komponent

5 POPIS SW VYBAVENÍ MODELU AUTA

Programové vybavení modelu auta bylo napsáno v programovacím jazyce C a vývojové prostředí byl použit CodeWarrior, který je postaven na základě vývojového prostředí Eclipse. [25]



Obrázek 18 Ukázka vývojového prostředí CodeWarrior

5.1 Použité knihovny

Rozšiřující modul FRDM-TFC s driversy pro ovládání motorů, výstupem pro servo a vstupem pro kameru má vlastní knihovnu, kterou poskytuje přímo firma NXP. Tato knihovna je složena z 12ti zdrojových souborů. Knihovna má funkce jako jsou například získání dat z kamery, funkce pro zatočení servomotoru o požadovaný úhel, funkce pro otáčení DC motoru žádanou rychlostí.

5.1.1 Funkce pro čtení dat z kamery

Funkce pro získání dat z kamery je zapsána ve zdrojovém souboru TFC_LineScanCamera.c v knihovně TFC pro desku FRDM-TFC. Prvně jsou nastaveny piny včetně analogového vstupu, pak se spustí hodiny a do SI vstupu kamery je poslán impuls pro vyslání dat. Data jsou ukládána do bufferu, z kterého jsou v hlavní funkci algoritmu vyčítána.

5.1.2 Funkce pro ovládání servomotoru

Pro servomotor je opět oddělený zdrojový soubor `TFC_Servo.c`. Ze začátku dochází k inicializaci, kdy se opět nastaví piny a spustí se hodiny. Pomocí PWM, což je pulzně šířková modulace, se vyšle signál do servomotoru pro zatočení o daný úhel. Výsledná funkce má tvar `voidTFC_SetServo(uint8_tServoNumber, floatPosition)`. `ServoNumber` udává připojení na desce FRDM-TFC, kde jsou dva konektory. `Position` se volí v rozsahu -1.0 do 1.0 pro otočení serva.

5.1.3 Funkce pro ovládání zadního pohonu

Stejně jako u předchozích funkcí i zde je oddělený zdrojový soubor `TFC_Motor.c`. Ze začátku dochází k inicializaci jako je nastavení pinů a spuštění hodin. Rychlost otáčení motoru je řešena pomocí PWM signálu. Výsledná funkce je ve tvaru `voidTFC_SetMotorPWM(floatMotorA, floatMotorB)`. První parametr `MotorA` zadává v rozsahu -1.0 až 1.0 pro pohon prvního motoru pro dopředný a zpětný chod. Druhý parametr `MotorB` má opět stejný rozsah a slouží pro dopředný či zpětný chod druhého motoru.

5.2 Vlastní řešení algoritmu

Na začátku hlavní funkce `main` dochází k inicializaci celé desky. Je volána funkce `TFC_Init()`, ve které se volají veškeré inicializační funkce včetně inicializace funkce pro kameru, servo i motory.

Následuje nekonečná smyčka pro chod programu. V podmínce se ověří `LineScanImageReady`, zda jsou data z kamery připravena v bufferu. Pokud data jsou připravena, tak ve `for` cyklu se projde celý buffer s daty z kamery a pokud je na nějaké pozici detekována černá čára, tak se ověří na kterém místě. V případě, že je černá čára detekována v první půlce bufferu, což je rozsah 23 až 64 (prvních 23 bitů zahazují z důvodu zkreslení), dojde k uložení informace o směru zatočení a pozici černé čáry. V opačném případě, že dojde k detekci černé čáry v druhé půlce bufferu, tj. rozsah 65 až 108 (posledních 20 bitů zahazují opět z důvodu zkreslení), dojde opět k uložení směru zatočení a pozici černé čáry do proměnných.

Po projití celého bufferu dat z kamery, tj. 128 bitů, se provede zatočení pomocí servomotoru, na základě získané informace o směru zatočení a na základě uložené informace o pozici černé čáry je vypočítaná žádaná pozice serva v rozsahu -1.0 až 1.0.

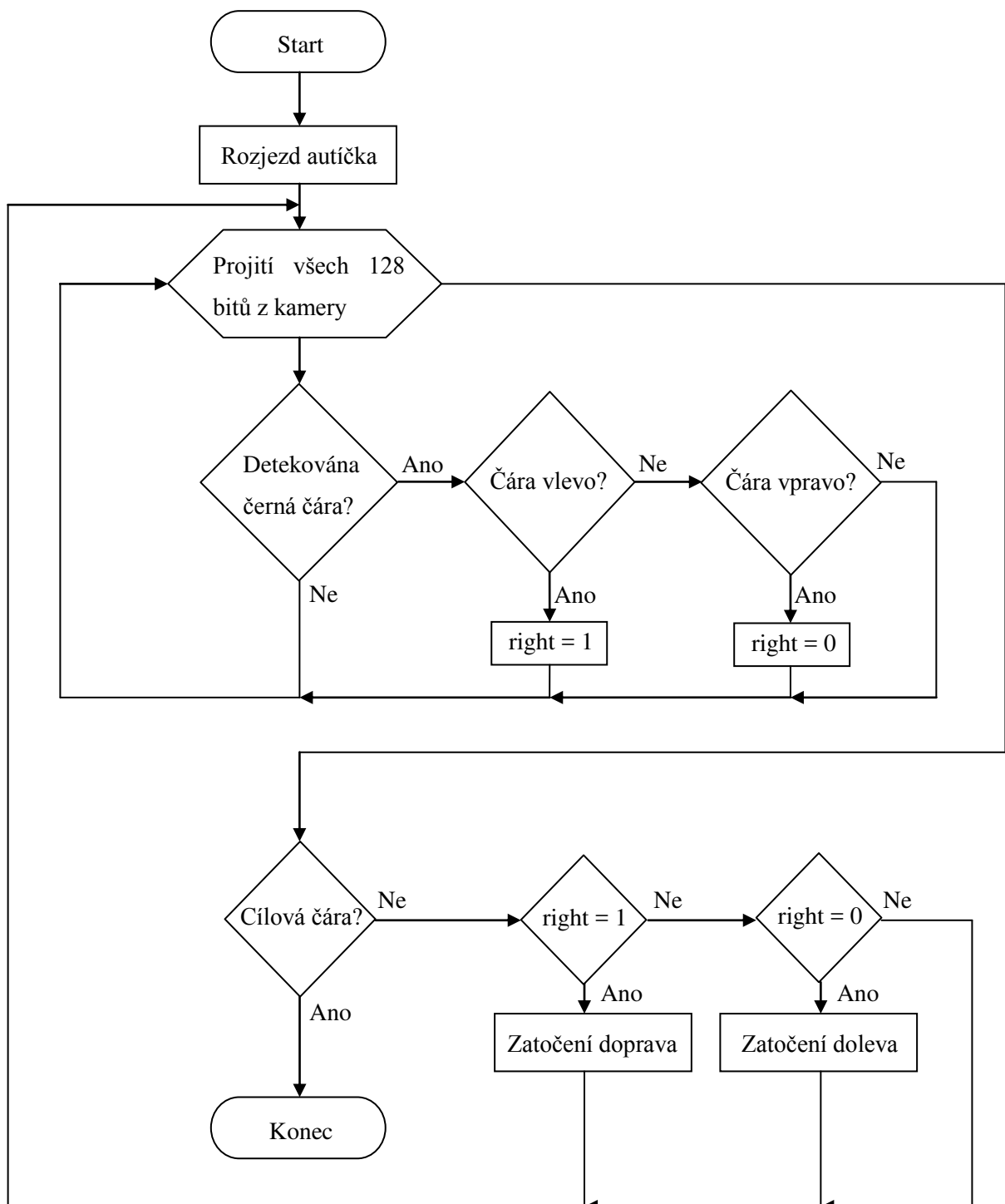
Pohon modelu je zajištěn po stisknutí tlačítka na desce a rychlost otáček je nastavitelná pomocí potenciometru umístěného na desce.[26]

V příloze je uveden zdrojový kód hlavní funkce `main`, ve které je řešen samotný algoritmus pro průjezd modelu auta dráhou.

6 ALGORITMUS PRO PRŮJEZD

Průjezd libovolnou dráhou musí být co nejvíce optimalizovaný, aby došlo k co nejrychlejšímu průjezdu. Je potřeba zvolit co nejlepší stopu průjezdu zatáčkami, aby nedošlo k opuštění dráhy.

Při rozjezdu modelu auta jede maximální, nastavenou rychlostí. Pokud dojde k detekci černé čáry na levé či pravé straně, dojde k zatočení opačným směrem. Narazí-li auto na cílovou čáru, dojde k detekci černé čáry na pravé i levé straně, model zastaví.



ZÁVĚR

Cílem této práce bylo sestavení a naprogramování modelu samořídícího vozidla. Vozidlo má za cíl projet předem neznámou trať ohraničenou černými čarami na okraji dráhy. Využívá k tomu optického snímače, který je zde realizován jednořádkovou kamerou, která má na výstupu obraz o velikosti 1x128px.

V teoretické části byly popsány autonomní vozidla současnosti. Základem samořídících vozidel je CPU, proto byli uvedeni vývojové kity a rozšiřující moduly, které k nim patří. V praktické části byl použit model auta určený pro soutěž The NXP Cup. Tento model je dodáván spolu s vývojovým kitem KL25Z, který byl popsán v praktické části a také s rozšiřujícím modulem FRDM-TFC. Model auta také obsahuje jako každý RC model servomotor pro zatáčení, 2 DC motory pro pohon, kde každý motor je umístěn na jednom zadním kole a také právě řádkovou kameru pro orientaci na dráze.

Model byl sestaven a zprovozněn. Programové vybavení bylo vytvořeno ve vývojovém prostředí CodeWarrior pomocí programovacího jazyka C. Byly využity i knihovny dodávané právě k desce FRDM-TFC od firmy NXP.

Práce může posloužit jako pomůcka pro případné zájemce, kteří se chtějí zúčastnit soutěže The NXP Cup. Zdrojové soubory mohou být inspirací pro vytvoření vlastního algoritmu pro průjezd neznámou dráhou.

SEZNAM POUŽITÉ LITERATURY

- [1] *Autonomní vozidla* [online]. [cit. 2016-05-23]. Dostupné z: <http://www.wisegeek.com/what-is-a-driverless-car.htm>
- [2] *Senzory autonomních vozidel* [online]. [cit. 2016-05-25]. Dostupné z: <http://www.computerworld.com/article/3005436/cybercrime-hacking/black-hat-europe-it-s-easy-and-costs-only-60-to-hack-self-driving-car-sensors.htm>
- [3] *Studie BI Intelligence* [online]. [cit. 2016-05-23]. Dostupné z: <http://www.businessinsider.com/report-10-million-self-driving-cars-will-be-on-the-road-by-2020-2015-5-6>
- [4] *Google cars* [online]. [cit. 2016-05-23]. Dostupné z: http://www.nytimes.com/2010/10/10/science/10google.html?_r=0
- [5] *Samořídící vozidlo od společnosti Google* [online]. [cit. 2016-05-25]. Dostupné z: <http://blog.americansafetycouncil.com/googles-self-driving-car-hit-a-bus/>
- [6] *Autopilot vozidel značky Tesla* [online]. [cit. 2016-05-23]. Dostupné z: <http://www.hybrid.cz/>
- [7] *Co je to Arduino* [online]. [cit. 2016-05-23]. Dostupné z: <https://www.arduino.cc/en/Guide/Introduction>
- [8] *Arduino produkty* [online]. [cit. 2016-05-25]. Dostupné z: <https://www.arduino.cc/en/Main/Products>
- [9] *Vývojové kity nxp kinetis* [online]. [cit. 2016-05-23]. Dostupné z: <http://www.nxp.com/products/software-and-tools/hardware-development-tools/freedom-development-boards:FREDEVPLA>
- [10] VÁŇA, Vladimír. *ARM pro začátečníky*. Praha: BEN - technická literatura, 2009. ISBN 978-80-7300-246-6.
- [11] GIBSON, J. R. *Arm assembly language: an Introduction*. Cambridge, UK: ARM Ltd, 2007. ISBN 978-184-7536-969.
- [12] *Architektura ARM Cortex* [online]. [cit. 2016-05-25]. Dostupné z: <https://indiatek.wordpress.com/2012/04/14/arm-cortex-a5-processor-overview/>
- [13] VALVANO, Jonathan W. *Embedded systems: Introduction to the Arm Cortex(TM)-M3 microcontrollers*. 2nd ed. s.l.: CreateSpace, 2012, xii, 462 s. ISBN 978-1477508992.
- [14] *ARM Cortex M* [online]. [cit. 2016-05-25]. Dostupné z: <http://www.arm.com/assets/images/tpl/compare-Cortex-M-diagramLG.png>

- [15] *Rozšiřující modul s LCD displejem* [online]. [cit. 2016-05-23]. Dostupné z: <http://www.tme.eu/cz/Document/277b52bb029c3584557815f229546df7/EA-LCD-004.pdf>
- [16] *Rozšiřující modul FRDM-FXS-MULT2-B* [online]. [cit. 2016-05-23]. Dostupné z: <http://www.nxp.com/products/software-and-tools/hardware-development-tools/freedom-development-boards/frdm-fxs-mult2-b-freedom-development-platform-for-freescale-sensors-with-bluetooth.:FRDM-FXS-MULT2-B>
- [17] *Rozšiřující modul FRDM-17510EJ-EVB* [online]. [cit. 2016-05-23]. Dostupné z: <http://www.nxp.com/products/power-management/engine-and-dc-motor-control/h-bridges/freedom-expansion-board-mpc17510-h-bridge-brushed-dc-motor-driver:FRDM-17510EJ-EVB>
- [18] *Robotické soutěže* [online]. [cit. 2016-05-23]. Dostupné z: <https://robotika.cz/competitions/cs>
- [19] *The NXP Cup* [online]. [cit. 2016-05-23]. Dostupné z: <https://community.freescale.com/docs/DOC-1284>
- [20] *Servo Futaba S3010* [online]. [cit. 2016-05-23]. Dostupné z: <http://www.servodatabase.com/servo/futaba/s3010>
- [21] *DC motory* [online]. [cit. 2016-05-23]. Dostupné z: <https://community.freescale.com/docs/DOC-93309>
- [22] *Line scan camera* [online]. [cit. 2016-05-23]. Dostupné z: <https://community.freescale.com/docs/DOC-1058>
- [23] *Rozšiřující modul FRDM-TFC* [online]. [cit. 2016-05-23]. Dostupné z: <https://community.freescale.com/docs/DOC-93914>
- [24] *Vývojový kit KL25Z* [online]. [cit. 2016-05-23]. Dostupné z: <https://developer.mbed.org/platforms/KL25Z/>
- [25] BARR, Michael a Anthony J MASSA. *Programming Embedded Systems: with C and GNU development tools*. 2nd ed. Sebastopol: O´Reilly, 2006, xxi, 301 s. ISBN 978-0-596-00983-0.
- [26] SMITH, Warwick A. *C programming for embedded microcontrollers*. 2nd ed. Susteren: Elektor International Media BV, 2008. ISBN 978-090-5705-804.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

SW	Software
HW	Hardware
LCD	Displej z tekutých krystalů
CPU	Centrální procesorová jednotka
GND	Uzemnění
PLA	Polyactid acid
DC	Stejnoseměrný proud
DIP	Vícenásobný přepínač zapouzdřený v keramickém či plastovém pouzdře
SRAM	Statická paměť
LED	Dioda vyzařující světlo
RGB	Červená – zelená – modrá
UART	Asynchronní sériové rozhraní
I2C	Dvojdřátová sériová sběrnice
SPI	Sériové periferní rozhraní
USB	Univerzální sériová sběrnice
NiMH	Nikl-metal hydridový akumulátor
PWM	Pulzně šířková modulace
AD	Analogově digitální převodník
DA	Digitálně analogový převodník
RC	Rádiem řízené

SEZNAM OBRÁZKŮ

<i>Obrázek 1 Senzory používané u autonomních vozidel [2]</i>	11
<i>Obrázek 2 Google samořídící vozidlo [5]</i>	14
<i>Obrázek 3 Seznam produktů Arduino [8]</i>	16
<i>Obrázek 4 Vývojový kit Arduino Leonardo [8]</i>	17
<i>Obrázek 5 Přehled procesorů architektury ARM [12]</i>	18
<i>Obrázek 6 Procesory s jádrem Cortex-M [14]</i>	19
<i>Obrázek 7 Rozšiřující modul s LCD displejem [15]</i>	20
<i>Obrázek 8 FRDM-FXS-MULT2-B od NXP [16]</i>	21
<i>Obrázek 9 FRDM-12510EJ-EVB od NXP [17]</i>	21
<i>Obrázek 10 Podvozek modelu auta</i>	25
<i>Obrázek 11 Použité servo v modelu autu [20]</i>	26
<i>Obrázek 12 Použitá kamera</i>	27
<i>Obrázek 13 Držák kamery</i>	27
<i>Obrázek 14 Rozšiřující deska FRDM_TFC</i>	28
<i>Obrázek 15 Vývojový kit FRDM-KL25Z [24]</i>	28
<i>Obrázek 16 Baterie pro napájení modelu</i>	29
<i>Obrázek 17 Blokové schéma zapojení komponent</i>	29
<i>Obrázek 18 Ukázka vývojového prostředí CodeWarrior</i>	30

SEZNAM PŘÍLOH

P I main.c

PŘÍLOHA P I: ZDROJOVÝ KÓD HLAVNÍ FUNKCE MAIN

```
#include"derivative.h"/* includeperipheraldeclarations */
#include"TFC\TFC.h"

voidstop_car(){
TFC_SetMotorPWM(0,0);
TFC_SetServo(0, 0);
}

intmain(void)
{
uint32_tt,i=0;

floatpoziceServa = 0;

int start = 0;
int pozice = 0;
intright = 0;
intfinish = 0;
int rozdeleni_log1_0;

TFC_Init();

for(;;)
{
TFC_Task();

if(TFC_PUSH_BUTTON_0_PRESSED){
start = 1;
}

if(TFC_PUSH_BUTTON_1_PRESSED){
start = 0;
TFC_SetMotorPWM(0,0);
TFC_SetServo(0, 0);
}

/* Při stisku tlačítka A na FRDM-TFC se model auta uvede do provozu */
if(start == 1){
TFC_HBRIDGE_ENABLE;
TFC_SetMotorPWM(TFC_ReadPot(0),TFC_ReadPot(0));

if(TFC_Ticker[0]>100 &&LineScanImageReady==1){
TFC_Ticker[0] = 0;
LineScanImageReady=0;
TERMINAL_PRINTF("\r\n");
TERMINAL_PRINTF("L:");

if(t==0)
t=3;
else
t--;

TFC_SetBatteryLED_Level(t);

/* Jelikož hodnoty z kamery jsou v rozsahu od 0 do cca 1500
* rozdělil jsem si tento rozsah na hodnoty log 0 a log 1.
```

```

* Zde pomocí potencimetru měním korekci tohoto rozdělení
  * kolem hranice 800.
*/
rozdeleni_log1_0 = 800 + (TFC_ReadPot(1)*100);

/* Projde se všech 128 bitů z kamery */
for(i=0;i<128;i++)
{
/* Ověření hodnoty zda-li je pod mým rozsahem pro log 1 */
if(LineScanImage0[i] <= rozdeleni_log1_0){
  TERMINAL_PRINTF("1"); //pouze informační zpráva do terminálu
/* Zda-li je log 1 nebo-li černá čára identifikována po levé straně
  * dojde k zatočení doprava (proměnná right).
  * Prvních 23 bitů ignoruji z důvodu zkreslení
  */
if(i > 23 && i < 64){
/* Detekce cílové čáry */
if(i > 30 && pozice == 0) finish = 1;
pozice++;
right = 1;
/* Pokud log 1 nebo-li černá čára je identifikována
  * na pravé straně, zatočíme doleva (right = 0).
  * Posledních 20 bitů ignoruji z důvodu zkreslení
  */
}elseif(i > 64 && i < 108){
/* Detekce cílové čáry */
if(right == 1 && pozice > 0){
  finish++;
}else{
  pozice++;
  right = 0;
}
}
/* Detekce log 0 nebo-li bílé čáry, pouze tisk informační
  * zprávy do terminálu
  */
}else{
  TERMINAL_PRINTF("0");
}

/* V proměnné "pozice" je uložena hodnota velikosti úhlu zatočení */

/* Po projití všech 128 bitů dat z kamery
  * dojde k zaovládání auta */
if(i==127){
/* Odřádkování v terminálu pro nová data z kamery */
  TERMINAL_PRINTF("\r\n");
/* V případě cílové čáry dojde k zastavení auta */
if(finish> 1){
  start = 0;
  stop_car();
}else{
/* Hodnota poslaná do funkce TFC_SetServo
  * musí být v rozsahu -1.0 do 1.0, proto je
  * zde převod na tento rozsah.
  * Hodnota 45 je odvozena z rozsahu detekce čáry
  * na levé či pravé straně (23 - 64, 64 - 108)
  */
  poziceServa = (float)pozice/45;

```

```
/* Zatočení doprava,  
* ošetření, maximální hodnota je 0.9  
*/  
if(right == 1){  
if(poziceServa > 0.9)  
TFC_SetServo(0, 0.9);  
else  
TFC_SetServo(0, poziceServa);  
/* Zatočení doleva,  
* ošetření, minimální hodnota je -0.9  
*/  
}elseif(right == 0){  
if(poziceServa > 0.9)  
TFC_SetServo(0, -0.9);  
else  
TFC_SetServo(0, poziceServa*(-1));  
}  
}  
finish = 0;  
pozice = 0;  
}  
}  
} //button  
}  
  
return 0;  
}
```