

VĚDECKÉ SPISY VYSOKÉHO UČENÍ TECHNICKÉHO V BRNĚ

*Edice Habilitační a inaugurační spisy, sv. 633*

*ISSN 1213-418X*

**Pavel Kučera**

**AUTONOMNÍ  
MECHATRONICKÝ SYSTÉM  
PRO UZAVÍRÁNÍ  
DIFERENCIÁLŮ VOZIDEL**

**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**  
**Fakulta strojního inženýrství**  
**Ústav automobilního a dopravního inženýrství**

**Ing. Pavel Kučera, Ph.D.**

# **AUTONOMNÍ MECHATRONICKÝ SYSTÉM PRO UZAVÍRÁNÍ DIFERENCIÁLŮ VOZIDEL**

**AUTONOMOUS MECHATRONIC SYSTEM  
FOR DIFFERENTIAL LOCKING OF VEHICLES**

**ZKRÁCENÁ VERZE HABILITAČNÍ PRÁCE**  
**OBOR KONSTRUKČNÍ A PROCESNÍ INŽENÝRSTVÍ**



**BRNO 2019**

## **KLÍČOVÁ SLOVA**

Autonomní mechatronický systém, vozidlo, diferenciál, MIL, HIL, jazyk C/C++, ECU, procesor STM32F4xx, Simulink, NI VeriStand, Atollic TrueSTUDIO, STM32 Cube MX

## **KEYWORDS**

Autonomous mechatronic system, vehicle, differential, MIL, HIL, language C/C++, ECU, processor STM32F4xx, Simulink, NI VeriStand, Atollic TrueSTUDIO, STM32 Cube MX

## **MÍSTO ULOŽENÍ PRÁCE:**

Habilitační práce je uložena v Areálové knihovně FSI VUT v Brně

## OBSAH

OBSAH.....	3
PŘEDSTAVENÍ AUTORA.....	4
ÚVOD.....	5
CÍLE PRÁCE.....	5
1 SOUČASNÝ STAV ŘEŠENÉ PROBLEMATIKY.....	6
2 VÝVOJ ŘÍDICÍHO ALGORITMU.....	7
3 VÝVOJ VÝPOČTOVÝCH MODELŮ PRO MIL TESTOVÁNÍ.....	13
4 MODEL IN THE LOOP - MIL TESTOVÁNÍ.....	14
5 HARDWARE IN THE LOOP - HIL TESTOVÁNÍ.....	16
6 VÝVOJ PROTOTYPU ECU.....	21
7 SOFTWAREOVÝ INTERFACE ECU.....	23
8 DISPLEJ.....	24
9 FINÁLNÍ TESTOVÁNÍ MECHATRONICKÉHO SYSTÉMU.....	25
ZÁVĚR.....	26
POUŽITÉ INFORMAČNÍ ZDROJE.....	27
ABSTRACT.....	32

## PŘEDSTAVENÍ AUTORA

**Ing. Pavel Kučera, Ph.D.**

Datum narození: 14. srpna 1986 v Poličce



### Přehled zaměstnání:

- 2018 - současnost Vysoké učení technické v Brně, Fakulta strojního inženýrství, Ústav automobilního a dopravního inženýrství, Brno, Pracovní pozice: Odborný asistent
- 2015 - 2018 Vysoké učení technické v Brně, Fakulta strojního inženýrství, Ústav automobilního a dopravního inženýrství, Brno, Pracovní pozice: Výzkumný pracovník
- 2010 - 2011 CAB Minerals s.r.o., Brno, Pracovní pozice: Konstruktor
- 2003 - 2010 Kovo HB s.r.o., Svatka, Pracovní pozice: Operátor,

### Dosažené vzdělání:

- 2011 - 2015 Ph.D., Fakulta strojního inženýrství, Brno, Obor/specializace: Konstrukční a procesní inženýrství
- 2009 - 2011 Ing., Fakulta strojního inženýrství, Brno, Obor/specializace: Automobilní a dopravní inženýrství
- 2006 - 2009 Bc., Fakulta strojního inženýrství, Brno, Obor/specializace: Strojní inženýrství

### Vědeckovýzkumná činnost a účast na projektech:

Zaměření autora v oblasti vývoje a výzkumu je v oblasti dynamiky vozidel, tj. vývoj mechatronických systémů, programování řídicích algoritmů, tvorba prototypů ECU, HIL testování, vývoj výpočtových modelů vozidel, MIL testování, vývoj simulátorů vozidla, konstrukce dílčích částí nebo celků, realizace měření dynamiky vozidla, vibrací, akustiky, zpracování naměřených dat atd. A to v rámci smluvního výzkumu např. s TATRA TRUCK a.s., ŠKODA AUTO a.s., ZETOR TRACTORS a. s. nebo v rámci projektů:

- 2019 - současnost TN01000026
- 2018 - současnost TAČR EPSILON IGW / TA137S02001
- 2018 - současnost TAČR EPSILON IGW / TA138S08000
- 2017 - současnost FSI-S-17-4104
- 2017 - současnost H2020 754869 EDEC
- 2016 - 2017 TE01020020
- 2013 - 2015 TE 01020020
- 2014 - 2016 FSI-S-14-2334
- 2013 FSI-J-13-2008, Modelování dynamiky vozidel II
- 2012 FSI-J-12-1803, Modelování dynamiky vozidla

### Pedagogická činnost:

- Semestrální projekt (QS3)
- Automobily (FAU)
- Vedení bakalářských, diplomových a doktorských prací
- Příprava nového předmětu QE1 - Systémy, řízení a mechatronika automobilů

## ÚVOD

Novodobé trendy směřují k zavádění autonomních mechatronických systémů a umělé inteligence prakticky do všech zařízení okolo nás. Řada vyvíjených a nabízených inteligentních mechatronických systémů souvisí s podvozkem vozidel a s řízením jejich hnacího traktu. I tato práce se zabývá vývojem systému spojeného s hnacím traktem vozidel. Jedná se o vývoj autonomního mechatronického systému pro automatické uzavírání diferenciálu nazvaného Autonomous control of the powertrain – ACP. Dále je vyvíjena nástavba tohoto systému pro sledování okolního prostředí, kde se vozidlo nachází a kam směřuje, s názvem Prediction of surroundings - PS.

Vyvíjený systém ACP-PS je sestaven z řídicí jednotky – ECU, senzorů a akčních členů. Základní funkcí systému je vyhodnocovat otáčky kol a z nich určovat, zda nedochází k prokluzu. Pokud prokluz nastane, řídicí systém vyhodnotí signály i z dalších senzorů, jako jsou pedály vozidla, snímač tlaku vzduchu v pneumatickém okruhu, signály z koncových spínačů v pneumatických válcích, kontrolních přepínačů a tlačítka, CAN zpráv z dalších ECU jednotek a dotykového displeje. Potom systém odešle signál akčnímu členu, který je sestaven z elektroventilu, pneumatického okruhu, pneumatického válce s koncovým spínačem a speciální zubové spojky. To znamená, že otevřením elektroventilu je puštěn tlakový vzduch do pneumatického válce, tím dojde k pohybu pístu s konzolou, která vyvodí sílu pro uzavření speciálních zubových spojek. Ty jsou umístěny v příslušných diferenciálech nebo použity pro spojení hnacích hřídelí předních náprav pro aktivaci pohonu všech kol. Řidič může systém ovládat přepínačem pro automatické a manuální řízení a dalšími přepínači a tlačítkem pro aktivaci pohonu všech kol, uzavření zadních osových diferenciálů a pro uzavření předních osových diferenciálů v manuálním režimu. Další možností ovládaní je nastavení jízdních módů řídicího systému pro silnici, polní cestu a terén/sníh na displeji, kde jsou zobrazovány i informace řidiči o stavu aktivace pohonu všech kol nebo uzavření příslušného diferenciálu.

Tato práce popisuje v první části stručný přehled používaných metod a nástrojů pro vývoj takovýchto mechatronických systémů včetně komerčně nabízených systémů pro řízení hnacího traktu vozidel. Dále je již popisován vlastní vývoj řídicího algoritmu, který byl i aplikován a odladěn pro vozy Tatra v rámci projektu TE 01020020 s názvem Centrum kompetence automobilového průmyslu Josefa Božka ve spolupráci se společností TATRA TRUCKS a.s., která ho implementovala do svých řídicích jednotek. V rámci vlastního vývoje a této práce bylo vytvořeno vlastní řešení s modernějšími prvky a nadstavbou pro sledování okolí vozidla. V práci je dále popisován vyvinutý nástroj pro tvorbu výpočtových modelů vozidel pro MIL testování, které bylo využito pro ověřování řídicího algoritmu. Dále je popsáno HIL testování na prototypch vozidel, návrh prototypu řídicí jednotky, programování grafického rozhraní pro komunikaci se systémem, programování dotykového displeje a závěrečné testování kompletního mechatronického systému, určeného již pro výrobu nulté série vozidel.

Tento systém je vyvíjen z důvodu lepšího využití hnacího momentu vozidla, šetření paliva a opotřebení pneumatik. Vozidlo vybavené tímto systémem je šetrnější k životnímu prostředí a k ochraně hnacího traktu před nevhodným ovládaním uzavírání diferenciálů nezkušenými řidiči. Proto je prioritně systém řízen automaticky.

## CÍLE PRÁCE

Cíle habilitační práce jsou vypsány v následujících bodech. Vývoj autonomního mechatronického systému pro uzavírání diferenciálu byl částečně navázán i na projekt TE 01020020 s názvem Centrum kompetence automobilového průmyslu Josefa Božka s konkrétním označením pracovního balíčku WP17. V rámci tohoto projektu a ve spolupráci se společností TATRA TRUCKS a.s. byl vyvíjený samostatný řídicí algoritmus odladěn pro jejich konkrétní typy nákladních vozidel a pro jejich vlastní elektronickou řídicí jednotku (ECU) vyvinutou dle požadavků algoritmu. Tento systém je veden ve společnosti pod názvem Tatra Drive

Managment - TDM. V rámci této práce a dřívějšího vývoje byl vyvinut kompletní mechatronický systém nazvaný Autonomous control of the powertrain – ACP s dále vyvíjenou nástavbou tohoto systému pro sledování okolního prostředí. Nástavba systému sleduje, kde se vozidlo nachází a kam směřuje s názvem Prediction of surroundings - PS. Systém ACP-PS je tedy koncipován modulárně pro velkou škálu typů vozidel. Hlavní cíle habilitační práce byly tedy:

- Vyvinout modulární řídicí algoritmus pro uzavírání diferenciálů různých typů vozidel
- Sestavit výpočtový model vozidla pro Model in the Loop - MIL testování
- Provést MIL testování vyvíjeného řídicího algoritmu
- Provést Hardware in the Loop – HIL testování vyvíjeného systému na prototypu vozidla
- Vyvinout řídicí jednotku pro tento systém
- Vyvinout softwarové rozhraní pro diagnostiku, nastavování parametrů, zavádění nového programu do procesoru ECU, testování a sběr dat z ECU
- Naprogramovat dotykový displej pro komunikaci uživatele s ECU.
- Provést testování kompletního vyvinutého autonomního systému.

## 1 SOUČASNÝ STAV ŘEŠENÉ PROBLEMATIKY

Pro vývoj mechatronických systémů mají firmy možnost využít řadu metod a nástrojů. Z hlediska vývoje řídicího algoritmu jsou používány základní jazyky jako je C/C++ [1], [2] a také dost často software Simulink [3]. Ten je užíván v různých nadnárodních firmách, jako je např. firma Bosch. Potom jsou využita různá vývojová prostředí, například Keil [4], IAR [5], TrueStudio [31] atd. Nyní se začínají využívat i inteligentnější jazyky jako je ADA [6]. S novodobým trendem autonomních systémů je také značně využíváno neuronových sítí pro řídicí algoritmy, které potřebují velice výkonný hardware např. pro zpracování obrazu, polohy atd.

Testování vyvíjených mechatronických systémů probíhá několika formami. První formou je Model in the Loop – MIL testování, následuje fáze Software in the Loop – SIL testování a dalším krokem je Processor in the Loop – PIL testování. Předposlední fází je Hardware in the Loop - HIL testování a poslední fází je závěrečné testování, které může odpovídat nějaké verzi nulté výrobní série. Pro jednotlivé fáze jsou používány především softwary od MathWorks – Simulink, National instruments – NI LabVIEW, NI VeriStand a hardware od National instruments [20], dSpace [16], NovaCarts [17], Vector [18] a Speedgoat [19]. [3], [15], [67], [68]

Pro vývoj prototypů ECU - Electronic control unit jsou používány různé softwary Allegro [7], EasyEDA [8], KiCad EDA [73], Eagle PCB [9], OrCAD [10], NI Multisim [11] atd. Pokud firma nechce věnovat investice do vývoje vlastního prototypu ECU, jsou k dispozici na trhu různé otevřené platformy ECU, to je např. FlexECU [12], Pi Innovo tzv. Open ECU [13], ECU od společnosti Bosch pro motorsport [14] atd.

Displeje pro vyvíjené mechatronické systémy si automobilka doprogramuje nebo nechá naprogramovat potřebou vizualizaci. Příkladem nabízených dotykových displejů mohou být produkty firem US Micro Products [21], Topcon Electronic [22] nebo Continental s 3D dotykovým displejem [23] a pro jednodušší aplikace např. mikrokontroléry s displejem od společnosti STMicroelectronics.

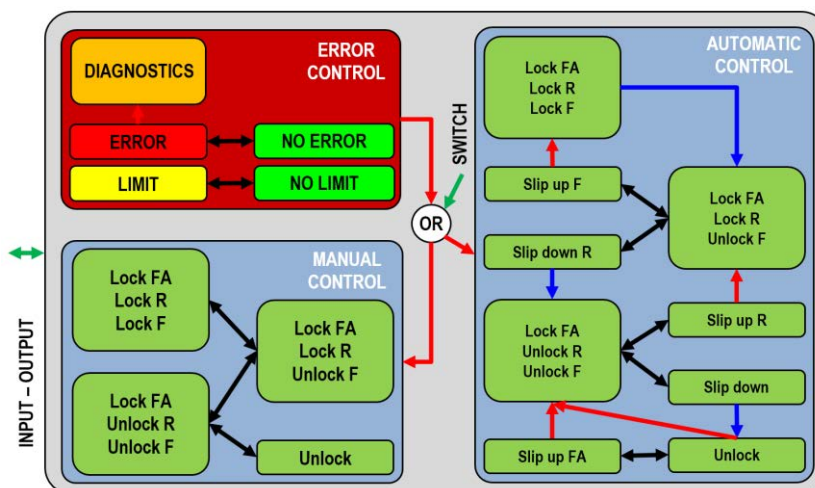
V posledním období je snahou sjednotit software nebo obecně protokol pro nastavování a sledování parametrů ECU při testování a vývoji. Proto určitě stojí za zmínku software ETAS – INCA [24], pracující podle norem ASAM [25].

V oblasti řízení hnacího traktu, uzavírání diferenciálů, aktivace pohonu všech kol a toku momentu, lze v oblasti nákladních vozidel zmínit systém pro uzavírání diferenciálů ZF ADM popsáný v [26] a [27], nebo systém od společnosti Meritor Driver-Controlled Differential Lock (DCDL) [28]. Jedná se tedy o systém, který vyhodnocuje prokluz kol a dle řídicího algoritmu je vyhodnocen stav, zda má být příslušný diferenciál uzavřen. Samozřejmě na trhu je řada systémů,

které řídí uzavírání diferenciálů u osobních vozidel, avšak tyto systémy v momentální chvíli není možné využívat u nákladních vozidel z hlediska vysokých přenášených hnacích momentů.

## 2 VÝVOJ ŘÍDICÍHO ALGORITMU

V rámci této kapitoly bude popsán vyvíjený řídicí algoritmus pro automatické uzavírání diferenciálů u osobních, nákladních a speciálních vozidel a traktorů. Cílem tohoto vývoje bylo vyvinout autonomní systém, který by uživateli ulehčil jízdu terénem a pomohl mu dopravit vozidlo do krajně nepřístupných míst, aniž by tam uvízlo. Aby mohl být vytvořen kompletní autonomní systém, bylo nutné naprogramovat řídicí algoritmus, kde jedno z možných schémat fungování je zobrazeno na následujícím obr. 1.



Obr. 1 Schéma řídicího algoritmu

Základní funkcí vyvíjeného autonomního mechatronického systému je vyhodnotit prokluz mezi jednotlivými koly a hřídelemi hnacího traktu určitého vozidla. Tím řídicí algoritmus určí podle nadefinovaného postupu, zda má příslušný diferenciál uzavřít nebo např. aktivovat pohon všech kol. Během vyhodnocování bere v úvahu další vstupy a faktory, které mohou zásadně ovlivnit rozhodování tohoto autonomního systému. Pokud systém rozhodne, že je potřeba uzavřít příslušný diferenciál, pošle signál akčnímu členu a ten příslušný diferenciál uzavře. Pokud je mechanická část diferenciálu skutečně uzavřena, systém je informován a může reagovat na případné další faktory určující, že vozidlo se nachází v nepříznivých adhezních podmínkách a je zapotřebí vhodně na to reagovat. Nebo naopak rozpozná stav, kdy je vhodné již příslušné diferenciály nebo pohon všech kol deaktivovat. Během těchto procesů zároveň probíhá diagnostika a systém určuje případnou závadu, aby se předešlo nebezpečným situacím. V případě potřeby je možné systém přepnout do tzv. manuálního režimu, kdy uživatel může diferenciály uzavírat jako u běžných sériových vozidel bez tohoto systému.

Aby bylo možné začít tvořit řídicí algoritmus, bylo zapotřebí vytvořit základní projekt obsahující periferie komunikující s jednotlivými piny procesoru. Procesor STM32F417 [29] byl využit pro sestavení prototypu ECU. Pro to byl využit STM32 Cube MX softwar [30], který slouží pro vygenerování základního projektu pro daný procesor v jazyce C a pro konkrétní vývojová prostředí, v tomto případě pro Atollic TrueSTUDIO software [31]. Pro komunikaci s periferiemi jsou použity HAL funkce popsané v UM1725 [32].

Jelikož se jedná o autonomní řízení uzavírání diferenciálu, je jedním z hlavních vstupů hodnota otáček kol, poloha pedálů, Controller Area Network - CAN zprávy z jiných řídicích jednotek – ECU, hodnota tlaku pneumatického okruhu, hodnota natočení volantu, poloha kontrolních prvků a



informace o fyzickém uzavření konkrétního diferenciálu. Výstupem je řízení akčních členů nebo signalizační kontrolky.

**Vstupy, CAN Komunikace a SPI Komunikace:** Mezi vstupy řídicího systému patří otáčky kol, poloha pedálů, tlak pneumatického okruhu, natočení volantu, kontrolní přepínače, spínač a displej a spínače uzavření diferenciálu. Dále je zde zahrnuta CAN a SPI komunikace.

**Otáčky kol:** Zpracování signálu nesoucího informace o otáčkách je předáváno ve formě digitálního signálu - DI. Kola vozidla jsou osazena indukčními snímači a tyto snímače generují specifický signál v závislosti na poloze vůči drázkám v náboji a tento analogový signál je konvertován do digitálního. Již tento signál ve formě digitálního signálu je zpracováván tak, že v procesoru je zastaven běžící program a provede se tzv. přerušení. V tomto případě je pin nastaven, že k přerušení dojde, pokud je detekována náběžná hrana vstupního signálu. Tohoto přerušení bylo využito z toho důvodu, aby hlavní program nezpomaloval zpracování otáček kol a tím především přesnost jejich hodnoty. Následně byla vyhodnocována perioda mezi jednotlivými přerušeními a přepočítána na otáčky. Pro případ vozidel, která mají nějaký typ brzdového mechatronického systému např. Anti-lock Braking Systém – ABS nebo Electronic Braking Systém – EBS, bylo využito možnosti brát informace o otáčkách z CAN zprávy Electronic Brake Controller 2 - EBC2, její označení je dáno normou J1939. Pokud vozidlo umožňuje využití této zprávy, je tím redukován počet užitých snímačů na vozidle. To souvisí s celkovou cenou systému i složitostí zástavby snímačů ve vozidle.

**Poloha pedálů:** Jedná se o hodnotu spojkového, brzdového a plynového pedálu, která je zpracována hardwarem ECU a procesoru předávána jako digitální signál. Tedy jednotlivé piny procesoru byly nastaveny na digitální vstupy a následně pomocí předpřipravené funkce HAL\_GPIO\_ReadPin() od společnosti STMicroelectronic [32] byla čtena hodnota daného pinu. U nových vozidel je předpokládáno získávání těchto hodnot z CAN zprávy motoru Electronic Engine Controller 2 - EEC2 a ve zprávě Cruise Control/Vehicle Speed – CCVS.

**Tlak pneumatického okruhu:** U vyvíjeného autonomního mechatronického systému je jednou z možností využití pneumatického okruhu pro uzavírání zubových spojek v diferenciálu a tím uzavření daného diferenciálu. Proto bylo nutné připravit i zpracování signálu ze snímače tlaku. Jelikož je předpokládáno, že takový snímač má analogový výstup, je nejdříve hardwarově upraven jeho rozsah a ochrana pro procesorový analogový vstup. Tomuto pinu musí být nastavena konfigurace v podobě analog-digitál – AD převodníku, dle informací v manuálu AN2834 na stránkách [29] a dle hodnot stanovených katalogovým listem procesoru STM32F417 na stránkách [32]. Potom je hodnota tlaku čtena pomocí základní funkce HAL\_ADC\_GetValue() a následně dále zpracovávána pomocí naprogramované funkce pro interpolaci a přepočítávána v algoritmu na konkrétní hodnotu tlaku.

**Natočení volantu:** Vyvíjený autonomní systém zpracovává především otáčky kol vozidla, aby vyhodnotil skluz. Pokud však vozidlo jede zatáčkou, algoritmus by vyhodnotil, že tam také vzniká skluz a uzavíral by diferenciály, aniž by to bylo nutné. Z toho důvodu je zapotřebí zpracovávat signál reprezentující natočení volantu vozidla. Proto byla vytvořena funkce pro jeho zpracování založena na teorii Ackermana, kdy lze přepočítat zatočení pravého nebo levého předního kola na teoretický skluz mezi jednotlivými koly nebo hřídelemi. Z hlediska signálů ze snímače natočení volantu se jedná o analogový signál a je zpracováván jako v přechodném odstavci.

**Kontrolní přepínače, spínač a displej:** Mechatronický systém je uživatelem ovládán pomocí tří přepínačů, jednoho spínače a displeje. První z přepínačů slouží pro nastavení automatického nebo manuálního režimu řízení uzavírání diferenciálů. Další dva přepínače jsou již pro manuální režim a to pro připojení pohonu všech kol a případných mezinápravových diferenciálů. Další je pro uzavření zadních osových diferenciálů. Tlačítko/spínač slouží pro uzavření všech předních osových diferenciálů vozidla. V tomto případě je signál pro procesor ve formě digitálního signálu a pomocí stejné funkce jako u pedálu je tato hodnota čtena z příslušného pinu. Displej je především určen pro informování uživatele, zda je např. aktivován pohon všech kol, nebo jaký

diferenciál je právě uzavřen. Také slouží pro nastavování řídicích módů ECU. To znamená, že pokud je řídicí systém nastaven na automatické řízení, může uživatel volit mezi třemi módy, které jsou silnice, polní cesta a terén/sníh. Komunikace probíhá pomocí CAN zpráv.

**Spínače uzavření diferenciálu:** Poslední ze zpracovávaných vstupů jsou spínače zpětných vazeb v pneumatickém válci, které informují řídicí algoritmus a uživatele o reálné aktivaci pohonu všech kol nebo uzavření příslušného diferenciálu. Opět se zde jedná o digitální signál, který je zpracován procesorem.

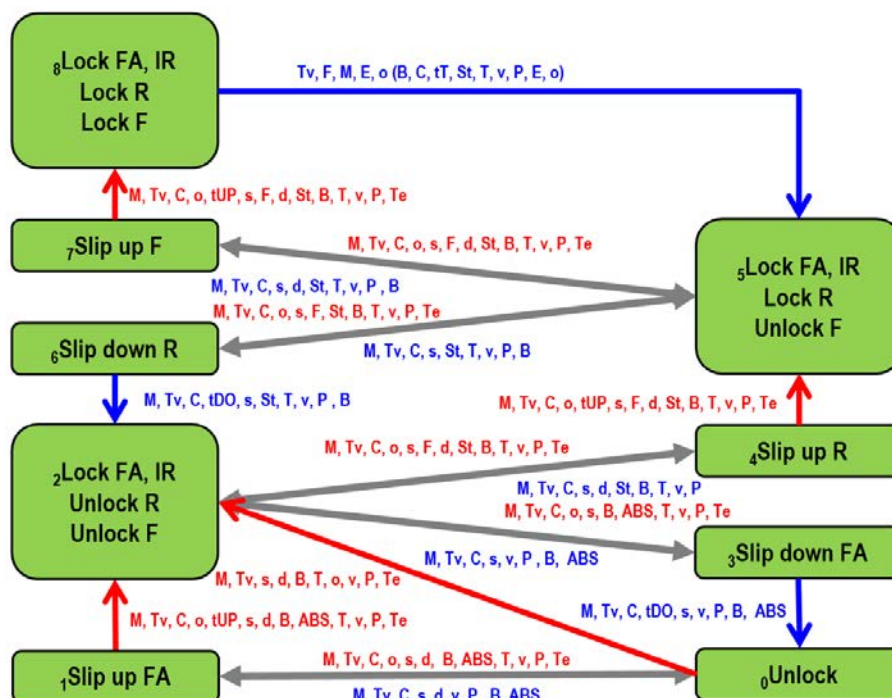
**CAN komunikace:** Jak již bylo zmíněno, některé vstupní signály pro řídicí algoritmus, jsou přenášeny pomocí CAN komunikace. Jelikož se jedná o aplikaci tohoto systému především pro nákladní vozidla a traktory, tak bylo vše programováno dle norem J1939 [33], [34], [36], [37] a [38]. Při programování kódů pro zpracování přijímání a odesílání CAN zpráv bylo především vycházeno právě z normy J1939 [33] a dalších odborných knih [39], [40]. Podrobnější popis fungování CAN komunikace je popsán právě ve zmíněných normách a nebude se podrobněji rozebírat. Zde bude popis zaměřen na CAN zprávy z různých ECU a na zprávy, které odesílá vyvíjená ECU. Jedná se tedy o hodnoty otáček ve zprávě EBC2, která má ID 0x18FEBF0B. Informace o stavu pedálů je ve více zprávách a vyvíjený autonomní systém používá zprávu EEC2 s ID 0x0CF00300 a CCVS s ID 0x18FEF100. Z první zprávy je zpracovávána hodnota Accelerator Pedal Position 1. Z druhé zprávy jsou zpracovávány data Brake Switch, Clutch Switch. Z této zprávy je zpracovávána ještě hodnota Parking Brake Switch. Aby nedocházelo při brzdění a aktivaci ABS k matení vyvíjeného systému, tak je čtena zpráva Electronic Brake Controller 1 - EBC1 s ID 0x18F0010B. Z jejích dat je zpracovávána hodnota Anti-Lock Braking (ABS) Active. K ochraně hnacího traktu souvisejících mechanických částí vyvíjeného systému je čtena zpráva s aktuální hodnotou hnacího momentu. To je ve zprávě Electronic Engine Controller 1 - EEC 1 s ID 0x0CF00400. Z dat zprávy je zpracovávána hodnota Actual Engine - Percent Torque. Pokud je tato hodnota k dispozici, algoritmus při překročení nastaveného limitu již neuzavírá diferenciály, aby nemohlo dojít k nadměrnému rázu v hnacím traktu a případnému poškození. U nákladních vozidel jsou další způsoby brzdění pomocí retardéru nebo motorové brzdy. Z toho důvodu jsou přijímány CAN zprávy Electronic Retarder Controller 1 - ERC1 s ID 0x18F00010 a 0x18F00029. Komunikace s displejem je přes CAN zprávy s ID 0x18EF09A7, která vysílá požadavek na změnu módu a po potvrzení zprávou ID 0x18EFA709, je zaslán konkrétní mód, ve kterém má fungovat autonomní řízení, tedy silnice, polní cesta nebo terén. Data pro informování řidičem na displeji jsou odesílána ve zprávě s ID 0x18FFCC09 s periodou 100 ms, kde jsou informace o aktivaci pohonu všech kol, uzavření jednotlivých diferenciálů, nastavený mód ECU, nastavení manuálního nebo automatického módu, chyb a limitů. Odesílané CAN zprávy pro testování a vývoj mechatronického systému jsou s ID 0x18FF1109, 0x18FF2109, 0x18FF3109, 0x18FF3209, 0x18FF4109, 0x18FF4209, 0x18FF4309, 0x18FF5109, 0x18FF45209, 0x18FF6109, 0x18FF6209, 0x18FF6309, 0x18FF7109 a jsou odesílány jen v případě přijímané zprávy ECU s ID 0x18FF81FA a příslušným kódem v jejich datech. Vyvíjený autonomní systém umožňuje aktivovat i naprogramovanou funkci pro omezování hnacího momentu nebo otáček motoru. Je to opět pro ochranu hnacího traktu jako v případě čtení aktuální hodnoty hnacího momentu motoru. Je zde využití CAN zprávy Torque/Speed Control 1 - TSC1. Základní nastavení pinů procesoru pro CAN komunikaci je dle manuálu RM0090 na stránkách [29]. Z hlediska kódu je pro příjem zpráv použita funkce HAL\_CAN\_RxFifo0MsgPendingCallback(), která je spuštěna při přerušení. To znamená, že když přijde nová CAN zpráva a není nastaveným filtrem hardwarově odfiltrována, dojde k přerušení a provede se další část kódu. Naopak pro odesílání CAN zpráv je používána funkce HAL\_CAN\_AddTxMessage(). Odesílání periodických zpráv je v jedné z hlavních pomalých smyček, realizovaných pomocí FreeRTOS.

**SPI komunikace:** Jak je popsáno v různých literaturách [41], [42] a především v manuálu pro daný procesor RM0090 na stránkách [29]. Serial Peripheral Interface - SPI komunikace slouží ke

komunikaci různých zařízení mezi sebou, kdy jedno zařízení je vždy master a ostatní jsou slave. Nastavení komunikace SPI pro procesor vyplývá z dokumentace pro užitý procesor. Pro přenos dat je používána základní funkce HAL\_SPI\_TransmitReceive(). Pomocí této komunikace jsou předávány v prototypu ECU data mezi dvěma moduly tj. flash paměť a MEMS obvodem obsahujícím tříosý gyroskop a akcelerometr. Tato externí paměť je užívána ve vyvíjeném autonomním systému pro ukládání chybových zpráv DM1, DM2 atd. dle normy J1939-73. Druhé zařízení je zde užito pro získávání sklonu vozidla pro příčný a podélný směr. Základem zpracování výstupních veličin je využití filtru popsaného na stránkách [43].

**Hlavní smyčka:** Vstupní hodnoty pro řídicí algoritmus jsou ve většině případů zpracovávány softwarově nebo přes různá přerušení pro rychlé děje. Pomalejší děje jsou rozděleny do dvou smyček. Jedna již byla zmíněna v kapitole o CAN komunikaci, ve které jsou odesílány CAN zprávy. Nyní bude popsána hlavní smyčka, ve které řídicí algoritmus autonomně rozhoduje o reakci celého vyvíjeného systému na aktuální podmínky o případném prokluzu kol. V první fázi jsou nejdříve načteny nastavitelné parametry. Také jsou nastaveny hodiny procesoru a jeho periférií, následuje nastavení digitálních pinů a smyčky pro kontrolu, zda je otočeno klíčkem zapalování. Pokud ano, následně jsou inicializovány nastavení pro individuální vstupy a výstupy procesoru. V neposlední řadě jsou spuštěny funkce časovače pro měření otáček kol a nastavení FreeRTOS, přes který běží i hlavní smyčka a obsahuje další funkce pro čtení digitálních a analogových vstupů, hodnot z MEMS modulu s gyroskopem a akcelerometrem, přijímání CAN zpráv od dalších ECU a displeje a především funkci pro řídicí algoritmus, kde je rozhodováno, zda bude aktivován pohon všech kol nebo bude uzavřen příslušný diferenciál.

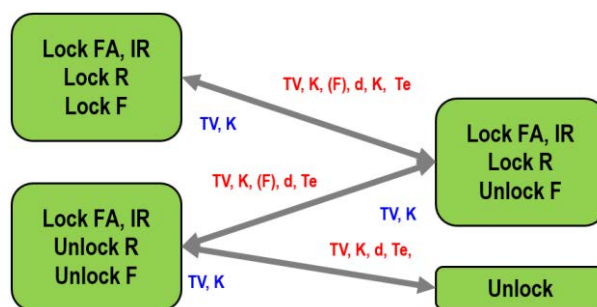
Součástí poslední funkce je zpracování již příslušných proměnných a následuje spuštění funkce Error() pro kontrolu chyb systému a funkce Algorithm(). V rámci vývoje byly testovány dva přístupy řídicího algoritmu. Jeden byl jako částečně nezávislý a druhý interně nazvaný závislé uzavírání diferenciálů. Ten druhý byl vybrán jako vhodnější a je zobrazen na obr. 2.



Obr. 2 Závislé řízení uzavírání diferenciálů

V první fázi je aktivován pohon všech kol a uzavření mezinápravových diferenciálů. V další fázi se uzavřou všechny zadní osově diferenciály a v poslední fázi jsou uzavřeny i všechny přední osově diferenciály. Než je spuštěna funkce Algorithm(), nejdříve je rozhodováno o tom, zda není v

systému nějaká porucha vyhodnocena ve funkci Error(), a následně dle výsledku je postupováno, což je popsáno v další podkapitole. V případě manuálního řízení probíhá uzavírání dle schématu, zobrazeného na následujícím obr. 3. Hlavní řídicí proměnou je hodnota vstupních přepínačů a spínače -  $K$  na palubní desce. Řidič v první fázi aktivuje pohon všech kol  $Lock FA$  a mezinápravové diferenciály  $Lock IR$ , potom uzavře zadní osově diferenciály  $Lock R$  a nakonec stiskne tlačítko pro uzavření předních osových diferenciálů  $Lock F$ . Dalšími proměnnými jsou velikost hnacího momentu motoru  $Te$ , kdy nesmí být překročena nastavená hodnota a hodnota rozdílu otáček  $d$  mezi hřídelemi, kde dochází k jejich spojování, tedy uzavírání diferenciálu. Tyto dva limity pro rozdíl otáček a hnací moment jsou aplikovány z důvodu bezpečnosti. Dalším řídicím parametrem pro manuální řízení je proměnná obsahující identifikátor typu vozidla  $TV$ . To znamená, zda se jedná o vozidlo s pohonem 4x4, 6x6, 8x8 nebo 10x10. Hodnota proměnné  $F$ , tedy zpětná vazba, reprezentuje reálný stav aktivace pohonu všech kol nebo uzavření individuálního diferenciálu. Ve schématu je v závorce, jelikož tuto funkci může mít i užitý pneumatický okruh ve vozidle. Jedná se o to, že pokud chce uživatel postoupit ve schématu směrem k uzavírání diferenciálů, musí být nejdříve aktivován pohon všech kol atd. Obecně k tomu schématu, pokud jsou proměnné označeny červeně, znamená to, že platí pro postup schématem směrem nahoru. Pokud jsou označeny modře, jedná o postup schématem dolů, tedy k deaktivaci pohonu všech kol a otevření všech diferenciálů. V daných buňkách je vždy zobrazeno to, co by mělo být v té poloze algoritmu uzavřeno nebo aktivováno.



Obr. 3 Manuální řízení uzavírání diferenciálu

Obecně by uživatel měl mít defaultně nastaveno automatické řízení, aby se o vše autonomně postaral řídicí systém. Manuální řízení by uživatel měl používat jen v případě, pokud je to nevyhnutelné nebo pokud systém hlásí poruchu a přepne řízení do manuálního režimu. Automatické řízení pracuje podle schématu na obr. 2. Tam jsou zobrazeny proměnné, podle kterých je vyhodnocována daná situace. To jsou typ vozidla -  $Tv$ , hodnota rozdílu otáček -  $d$ , procentuální hodnota hnacího momentu motoru -  $Te$ , zpětná vazba informující o stavu pohonu všech kol a diferenciálů -  $F$ , hodnota brzdového pedálu -  $B$  (pokud je jakkoliv brzděno v řídicím schématu je postupováno směrem dolů), hodnota spojkového pedálu -  $C$ , hodnota plynového pedálu -  $T$ , error - porucha systému -  $E$ , hodnota aktivace ABS -  $ABS$ , mód nastavení (silnice, polní cesta, terén/sníh) -  $M$ , otáčky kol/hřídelí dle polohy ve schématu řídicího algoritmu -  $o$ , tlak v pneumatickém okruhu -  $P$ , skluz mezi jednotlivými koly nebo hřídeli -  $s$ , úhel natočení předních kol -  $St$ , čas, po který řídicí algoritmus čeká, než může sestoupit ve schématu -  $tDO$ , čas testování, po kterém řídicí algoritmus otevře přední diferenciály -  $tT$ , čas, po který řídicí algoritmus čeká, než může vystoupit ve schématu nahoru -  $tUP$  a rychlost vozidla -  $v$ . Samozřejmě k daným proměnným je vždy přiřazen nějaký limit nebo hodnota, vůči které jsou porovnávány.

Nyní bude popsána základní filozofie fungování řídicího algoritmu. Schéma je rozděleno na 9 úrovní, které odpovídají např. aktivaci pohonu všech kol  $LockFA$ , nebo uzavření příslušných diferenciálů, případně čekání, zda situace prokluzu přetrvává určitý časový interval, aby mohlo být ve schématu postupováno výše. V první fázi je pohon všech kol deaktivován a jednotlivé diferenciály otevřeny. Pokud vozidlu začne v nepříznivých adhezních podmínkách prokluzovat

kolo a vyhodnocený slip/prokluz je nad stanovený limit, je postoupeno ve schématu algoritmu na úroveň 1 „Slip up FA“. Aby opravdu bylo ve schématu opravdu přežito na tuto úroveň, je potřeba splnit další podmínky. Pokud je vše splněno, řídicí algoritmus postoupí ve schématu do bloku „Slip up FA“. Tam probíhá další kontrola, a pokud je splněna, je ve schématu postupováno na úroveň 2, kde již je odeslán výstupní signál akčnímu členu pro aktivaci pohonu všech kol a případně uzavření mezinápravových diferenciálů. Pokud v rámci čekání není nějaká podmínka splněna, tak se algoritmus vrací na nultou úroveň ve schématu. Druhou možností jak se dostat na úroveň 2 je, že algoritmus rozpozná, že se vozidlo rozjíždí do kopce. Tím je rozložen hnací moment a nejsou zatěžovány jen individuální hřídele a je zabráněno jejich namáhání. Pro další postup ve schématu na úroveň 4 musí být splněny další podmínky. S postupem na jednotlivé úrovně souvisí i nastavení módů, pokud je nastaven mód silnice, řídicí algoritmus postupuje ve schématu maximálně do úrovně 2. Pokud je nastaven mód polní cesta, tak je postupováno maximálně na úroveň 5 a pokud je nastaven mód terén/sníh, tak řídicí algoritmus může procházet kompletním schématem. Nyní se vrátíme k popisu postupu na úroveň 4, pokud jsou podmínky splněny, postupuje se výše. V případě, že nejsou splněny podmínky, sestupuje se v algoritmu dolů přes úroveň 3 až na 0. Pokud je algoritmus na úrovni 4, jsou sledovány signály zase po určitý časový interval. Při splnění je postoupeno na úroveň 5 nebo naopak je sestoupeno na úroveň 2. Z úrovně 5 jsou zachovány stejné postupy pohybu ve schématu jen s odlišným nastavením limitních parametrů. V případě, že nepříznivé adhezní podmínky přetrvávají dál, postupuje algoritmus až na úroveň 8, která je vrcholem schématu. V tu chvíli je aktivován pohon všech kol a jsou uzavřeny všechny diferenciály. Zároveň algoritmus nemá žádné signály, podle kterých by určil, zda prokluz vozidla stále probíhá, z toho důvodu je zde naprogramována tzv. testovací smyčka. Pokud některá jiná hodnota neurčí, že řídicí algoritmus sestoupí na úroveň 5, tak je čekáno po stanovený časový interval a potom se přechází na úroveň 5 testovací smyčkou. Následně je vyhodnocen stav, a pokud dochází stále k prokluzu vozidla, pokračuje se zase nahoru až na úroveň 8, pokud není, tak je ve schématu pokračováno směrem dolů a tím je předpokládáno, že vozidlo např. projelo bahnitým terénem a nyní již zásah systému není nutný. Další probíhající funkce po projití jednoho cyklu v rozhodovacím mechanismu jsou funkce pro nastavení výstupních pinů a tím aktivaci akčních členů a funkce pro naplnění dat CAN zpráv.

**Chyby a limity:** V hlavní smyčce FCE\_Main\_Loop() se nachází hlavní část řídicího algoritmu s názvem funkce Loop\_Algorithm(), která dále obsahuje další funkce popsané v předchozí kapitole. Také se tam nachází funkce Error(). Tato funkce předchází funkci Algorithm() a je to z toho důvodu, aby se prvně vyhodnotily případné poruchy řídicího systému a následně se podle toho řídicí algoritmus zachoval. Nejvyšší prioritu mají poruchy, při kterých je systém deaktivován v obou režimech řízení (automatické, manuální). Poruchy ve druhé oblasti jsou, kdy systém zůstává funkční v manuálním režimu, ale v automatickém režimu je deaktivován. A poslední oblast z poruch je ta, kde poruchy jsou detekovány, ale systém není v obou režimech nijak omezen. Z hlediska řídicího algoritmu jsou také kontrolovány limity pro aktivaci akčních členů, aby nedošlo k destrukci hnacího traktu a to i v případě manuálního řízení, kdy zkušenosti řidičů jsou v dnešní době dost často nedostatečné. Proto je řidič informován o případných limitech.

Užitím daných funkcí pro vyhodnocení poruch nebo limitů jsou dané výstupy zapsány do jednoho vektoru pro další vyhodnocení a následně ukládány do externí paměti a reprezentují aktivní a dříve aktivní poruchy DM1 a DM2. Zpracované poruchy a limity jsou odesílány pomocí CAN zprávy také displeji. To znamená, že na displeji je zobrazeno, že systém je v pořádku, dochází k nějakému limitu, nebo má systém poruchu.

**Výstupy:** Jedná se o digitální výstupy, které ovládají výkonné elektronické součástky. Hodnota tohoto výstupu z procesoru je nastavována pomocí předdefinované funkce HAL\_GPIO\_WritePin(). Ovládané elektronické součástky zajišťují ovládání akčních členů. Je užito akčního členu ve formě elektroventilu, který pouští tlakový vzduch do pneumatického válce pro mechanické ovládání zubových spojek.

**Bootloader:** Bootloader je tzv. zařízení nebo program pro zavádění hlavního programu obsahujícího řídicí algoritmus a vše s ním související. Při vývoji a běžném provozu bylo použito nahrávání programu užitím dvou přístupů. Prvním přístupem je připojení hardwarového bootloadru k pinům procesoru k tomu určených. V tomto případě bylo využito programování procesoru pomocí Serial Wire Debugging – SWD, kdy potřebné piny byly nastaveny v projektu a pomocí hardwaru ST-Link [44] nebo J-Link [45] byl program implementován do procesoru. Jako druhý způsob byl naprogramován tzv. zaváděcí program. Ten je umístěn na začátku paměti procesoru a byl naprogramován tak, aby reagoval na komunikaci CAN. To znamená, že uživatel je schopen zavést novou verzi hlavního programu pomocí CAN komunikace do paměti procesoru. K tomuto účelu bylo naprogramované diagnostické rozhraní komunikující s hardwarem PCAN [46]. Tento systém odešle novou verzi hlavního programu do procesoru, který ji uloží dle definice na příslušné místo v jeho paměti a spustí novou verzi programu.

### 3 VÝVOJ VÝPOČTOVÝCH MODELŮ PRO MIL TESTOVÁNÍ

Aby bylo možné testovat vyvíjené řídicí algoritmy, prochází vývojem i vlastní knihovna obsahující bloky s matematickými modely zastupujícími jednotlivé části vozidel. Pro tvorbu bloků bylo vycházeno z literatury [63], [64] a [65], kde je popsána tvorba modelu v softwaru Simulink a obecný matematický popis částí hnacího traktu. Zde budou popsány jen základní funkce knihovny s bloky, jelikož to není hlavním výstupem této práce a podrobnější popis je v doktorské práci [47] nebo ve vydaných článcích [48], [49], [50], [51], [52], [53], [54], [55] a [56].

Knihovna je tvořena v prostředí softwaru Simulink. Vytvořená rozsáhlá knihovna s bloky je určena pro sestavování výpočtových modelů vozidel a pro simulaci jejich dynamiky, řešení vibrací a především pro testování vyvíjených mechatronických systémů. Celá knihovna se dělí na další podknihovny.

**Knihovna ECU:** Jedná se o knihovnu, která obsahuje bloky reprezentující různé vyvíjené řídicí algoritmy. Řídicí algoritmy, včetně zde popisovaného, jsou programovány v jazyce C, C++, nebo v softwaru Simulink.

**Knihovna funkcí:** Tato knihovna obsahuje různé funkce používané nejenom pro řídicí algoritmy, ale také pro různé výpočtové modely fyzikálních systémů. Jako jsou různé tabulky, průměrování hodnot vektorů atd.

**Knihovna Multibody:** Knihovna multibody je ve fázi vývoje.

**Knihovna sensorů:** V této knihovně jsou bloky popisující chování různých snímačů, aby bylo možné generovat obdobné signály jako v reálném provozu.

**Technická knihovna:** Obsahuje bloky pro popis mechanických částí jako je hřídel, rotační kotouč, základní převod, zubové spojky atd. Tyto bloky jsou základem torzních systémů a bloků pro sestavování hnacího traktu vozidel.

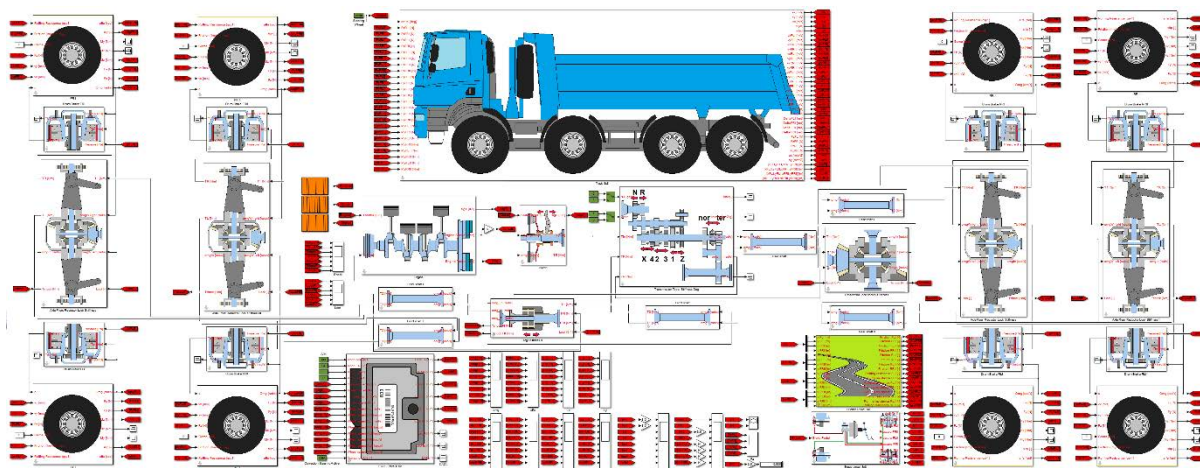
**Knihovna pro vozidla:** Tato knihovna obsahuje bloky reprezentující různé typy náprav. Dále jsou zde zahrnuty bloky diferenciálů a jsou zde zahrnuty bloky reprezentující jednoduchý zubový převod až pro celé převodovky různých výrobců jako je Tatra nebo ZF. Dále je zde podknihovna s brzdami, která matematicky popisuje bloky pro bubnové a kotoučové brzdy. Následuje podknihovna s bloky vozidla. V této knihovně jsou bloky pro různé typy vozidel např. nákladní nebo terénní čtyřkolky. Tyto bloky řeší podélnou a příčnou dynamiku vozidla. Nedílnou součástí dalších podknihoven jsou třecí spojky a zubové spojky. Matematický popis třecích spojek vychází ze dvou přístupů tj. podle Karnoppa [57] nebo vlastním přístupem. Podknihovna s bloky motorů umožňuje simulovat chování reálného motoru a dle potřebné složitosti a věrohodnosti je vytvořeno několik verzí včetně dílčích částí. Úzce spojené jsou podknihovny pro popis vozovky a pneumatik. Tím jsou definované adhezní podmínky a chování pneumatik dle zadaných parametrů. Rovnice pro vytvoření bloku pneumatiky byly čerpány z literatury [58] a [66], kde autor Pacejka popisuje chování pneumatiky z hlediska sil a momentů pro všechny směry.

Kombinací zmíněných bloků je schopen uživatel sestavit jim požadovaný typ vozidla a po nastavení parametrů může simulovat a testovat jízdní situace.

#### 4 MODEL IN THE LOOP - MIL TESTOVÁNÍ

Model in the Loop testování je založeno na spojení výpočtového modelu např. vozidla a vyvíjeného řídicího algoritmu. Toto spojení je realizováno v uzavřené smyčce, kdy jsou z výpočtového modelu vozidla předávány informace řídicímu algoritmu a naopak řídicí algoritmus předává informace výpočtovým modelům akčních členů na modelu vozidla. To má zásadní výhodu, že není potřeba pro prvotní testování vyvíjeného řídicího algoritmu žádný prototyp vozidla, ECU, nebo snímačů a akčních členů. Další velice významnou výhodou je, že v této fázi vývoje je obecně nalezeno nejvíce možných chyb např. způsobených programováním nebo filozofií fungování celého systému. Také nemůže dojít k destrukci částí vozidla a mohou být testovány různé atypické jízdní situace, aniž by byl ohrožen řidič nebo obsluha. Všechny tyto výhody vedou k jednomu cíli, tj. ušetřit čas a finanční náklady při vývoji, a proto je tento způsob testování při vývoji mechatronických systémů významně užíván.

Dříve popsaný vyvíjený řídicí algoritmus byl touto formou také testován. Pro tento účel byly sestaveny výpočtové modely vozidel pomocí vlastní knihovny s bloky popsané v předchozí kapitole. Jedná se o tři typy vozidel s pohonem 4x4, 6x6 a 8x8. K těmto modelům byl připojen blok obsahující vyvíjený řídicí algoritmus a tím to vytvořilo smyčku pro MIL testování. Výpočtový model vozidla 8x8 je zobrazen na následujícím obrázku.

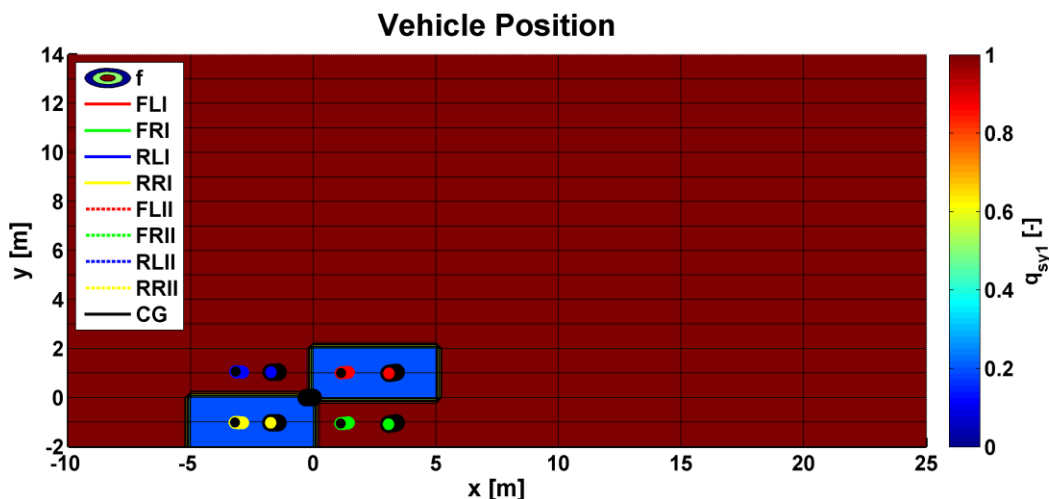


Obr. 4 Výpočtový model vozidla s pohonem 8x8 pro MIL testování

**Simulace:** Po sestavení výpočtových modelů mohl být vyvíjený řídicí algoritmus testován, aniž by byla potřeba nějaký prototyp vozidla nebo ECU. Tím mohl probíhat paralelní vývoj a byl ušetřen čas. Proto byly připraveny různé simulace pro MIL testování z hlediska jízdy vozidla. Testovalo se rozjíždění, zatáčení, brzdění, přerazování rychlostních stupňů, a to samozřejmě při různých adhezních podmínkách, aby docházelo k prokluzu kol a tím řídicí algoritmus musel reagovat na signály z výpočtového modelu vozidla. Po vyhodnocení situace odesílal signály pro aktivaci akčních členů a tím se aktivoval pohon všech kol a uzavíraly se příslušné diferenciály. Kladným výsledkem bylo samozřejmě, když vozidlo projelo simulovaným terénem, aniž by tam uvízlo. Z výstupních dat byly vykresleny vždy potřebné grafy a probíhala vizuální kontrola, zda řídicí algoritmus reagoval na vstupní signály dle požadavků. V případě, že se našla chyba nebo algoritmus reagoval nevhodným způsobem, byla provedena změna kódu a MIL testování se spustilo znovu.

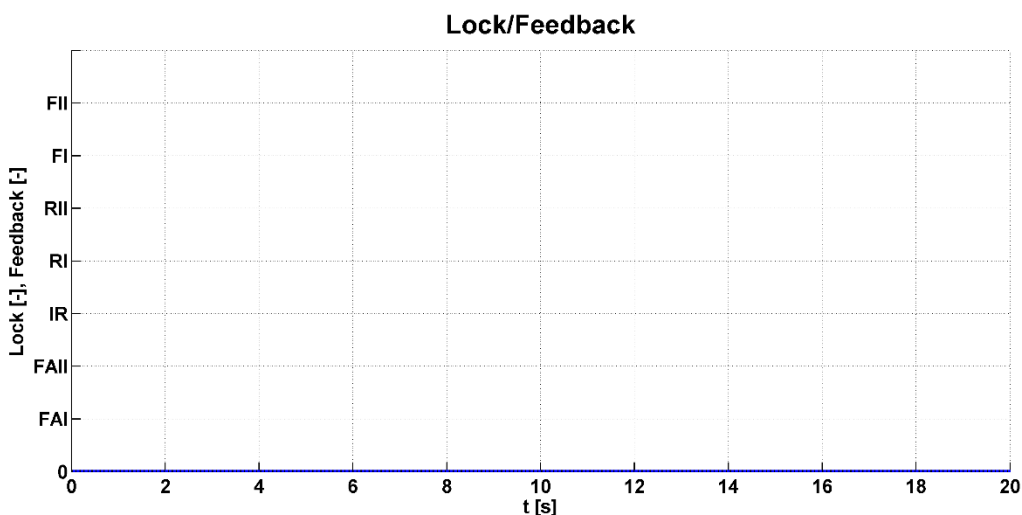
Ukázka jednoho MIL testování je zobrazena na následujících obrázcích, kdy další výsledky byly publikovány v různých časopisech [60], [61] [62] a [62]. Na následujícím obrázku je

zobrazen graf reprezentující polohu vozidla a jeho kol. Zároveň je plocha grafu v určitých místech podbarvena různými barvami. Tyto hodnoty zastupují měřítko součinitele přilnavosti mezi pneumatikou a vozovkou. To znamená, že vozidlo s pohonem 8x8 bylo v první fázi testování stojící pravými koly zadních náprav v bahnitěm terénu a levými koly přední nápravy také v bahnitěm terénu ale s jinými parametry. První testování je pro nastavení manuálního řízení, kdy nebyl manuálně aktivován pohon všech kol ani uzavřen žádný diferenciál.



Obr. 5 Poloha vozidla během MIL testování a při nastaveném manuálním řízení

Jelikož při simulaci rozjezdu některá kola byla v bahnitěm terénu a ostatní na asfaltu, došlo k prokluzu pravých zadních kol. Ostatní kola měly hodnotu otáček nulovou včetně předních kol, jelikož v první fázi jsou poháněny u výpočtového modelu vozidla jenom zadní nápravou. Tím vozidlo zůstalo stát na místě. Aby řidič z tohoto bahnitěho terénu vyjel, musel by přepnout přepínač pro aktivaci pohonu všech kol.

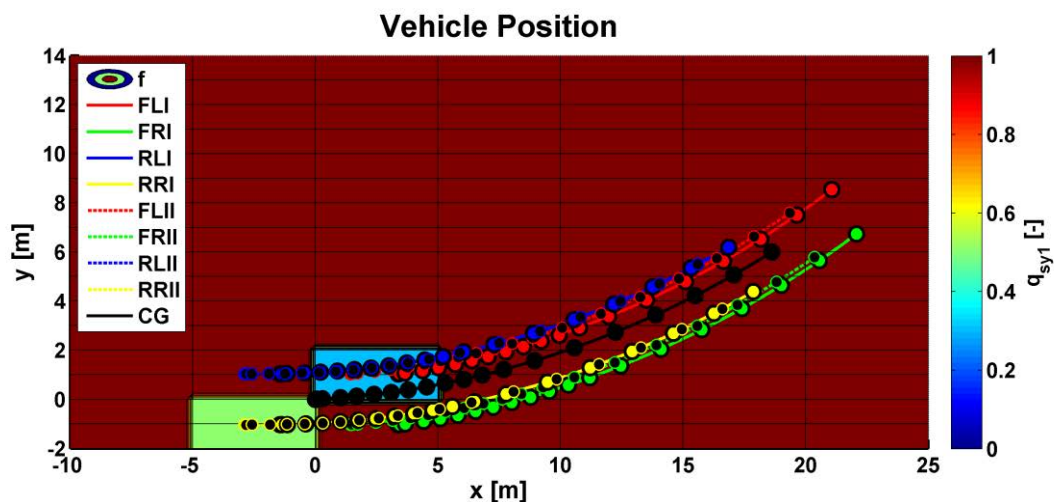


Obr. 6 Průběh signálů pro akční členy (černá barva) a zpětné signály koncových spínačů (modrá barva) při MIL testování a nastaveném manuálním řízení

Pokud by to nebylo dostačující, tak by následně musel zmáčknout přepínač pro uzavření zadních osových diferenciálů a podržet tlačítko pro uzavření předních diferenciálů. Ale jak je vidět na výsledcích obr. 6, všechny signály  $Lock_i$  jsou nulové, tedy řidič nic neaktivoval. Simulace dále probíhala bez výsledku rozjetí vozidla. Ukázka druhého MIL testování je na následujícím obrázku,

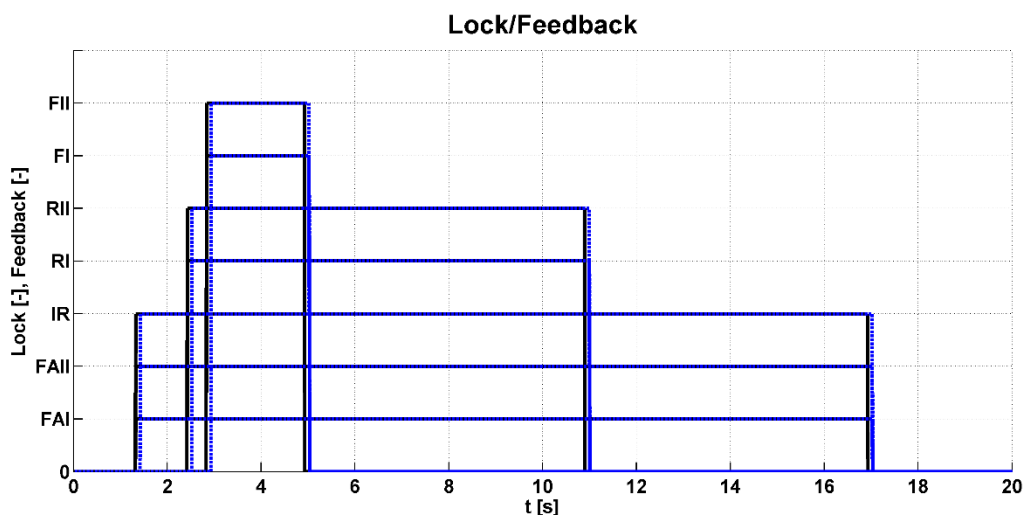


kde již byl mechatronický systém nastaven na automatické řízení a všechna ostatní nastavení z hlediska signálů pedálů nebo zatočení volantu zůstala stejná.



Obr. 7 Poloha vozidla během MIL testování s nastaveným automatickým řízením

Předchozí graf polohy vozidla určuje, že algoritmus vyhodnotil situaci při rozjezdu vozidla a jeho jízdy terénem a postupně aktivoval pohon všech kol, nebo uzavíral příslušné diferenciály. Tím se vozidlo postupně dostalo z bahnitého terénu bez zásahu řidiče, dokud všechna kola nebyla na asfaltu. To je vidět i ze signálů  $Lock_i$  na obr. 8, kdy po časovém úseku 17 s již řídicí algoritmus na situaci nemusí reagovat. Při tomto testování bylo např. ověřeno, že řídicí algoritmus reaguje na vstupní signály a uzavírá příslušné diferenciály. Naopak je kontrolováno, že vypne akční členy systému po skončení nepříznivých adhezních podmínek, ale také je kontrolována správná funkce korekce skluzu při zatáčení. Pokud by tato korekce nefungovala při MIL testování, algoritmus by zasílal signál akčním členům i v případě, když by celé vozidlo bylo na asfaltovém povrchu.



Obr. 8 Průběh signálů pro akční členy (černá barva) a zpětné signály koncových spínačů (modrá barva) při MIL testování a nastaveném automatickém řízení

## 5 HARDWARE IN THE LOOP - HIL TESTOVÁNÍ

HIL testování je využíváno za účelem testování na prototypch vozidel nebo jiných zařízeních, aniž by bylo nutné mít již vyroben prototyp ECU. To nese značné výhody z hlediska ušetření vývojového času a především jeho finanční náročnosti. To znamená, že hardware pro HIL je využit jako náhrada řídicí jednotky, kde je implementován vyvíjený řídicí algoritmus. Ten je již

odladěn z MIL testování. Nyní je testovací smyčka již složena z hardwaru, elektroniky vozidla, akčních členů, snímačů a mechanických částí. V podstatě je sestaven prototyp vozidla, který se při testování doladuje k nulté výrobní sérii. Prototyp vozidla má zakomponovány všechny prvky mechatronického systému krom prototypu ECU, kdy náhradou je právě hardware pro HIL testování. Jak bylo zmíněno nejvíce možných chyb, které vzniknou při vývoji, je nalezeno při MIL testování a při HIL testování je předpokládán také určitý rozsah možných chyb, ale již v mnohem menším měřítku. Potom hledání chyb systému při testování na kompletním prototypu vozidla je již zanedbatelné.

Takže z hlediska hardwaru bylo použito zařízení od společnosti National Instruments. To bylo spojeno s prototypem vozidla určeného na testování mechatronického systému pro uzavírání diferenciálů. Další možností, která je využívána, je vývoj Real-time simulátoru vozidla více popsán v [47]. Tento simulátor využívá sestavené výpočtové modely vozidel. Vozidlo lze řídit v reálném čase pomocí herního volantu a pedálu a uživatel vidí pohyb vozidla včetně palubní desky v naprogramovaném grafickém rozhraní. Z hlediska vývoje mechatronického systému je výhodou tohoto simulátoru, že je využíván pro testování vyvíjeného prototypu ECU s reálnými zásahy řidiče.

**Hardware:** Jak již bylo zmíněno pro HIL testování je používán hardware od společnosti National Instruments a z jejich dokumentace na internetových stránkách [67] byly čerpány základní informace. Základem je hardware NI 3110RT, kde je umístěn procesor pro implementaci vyvíjeného řídicího algoritmu, který byl v tomto případě z C/C++ jazyka nebo softwaru Simulink kompilován do příslušného formátu dll knihovny. K tomuto hlavnímu hardwaru jsou připojeny dva sloty pro umístění modulů se vstupy a výstupy. Jedná se o sloty s označením NI 9159 a NI 9157. Pro danou aplikaci byl použit první slot s programovatelným hradlovým polem Field Programmable Gate Array – FPGA. V tomto hradlovém poli je naprogramování komunikace jednotlivých modulů s hlavním hardwarem, ale také to uživatel může doplnit vlastním kódem. To je využíváno pro zpracování rychlých dějů a v případě testování vyvíjeného systému zde bylo naprogramováno zpracování otáček vozidla obdobně jako při zpracování procesorem, popsán v dřívějších kapitolách.



*Obr. 9 Implementovaný hardware v prototypu vozidla pro HIL testování*

Pro HIL testování vyvíjeného systému, byly použity následující moduly. Dva moduly NI 9229 pro snímání signálů z indukčních snímačů otáček kol, jeden modul NI 9239 pro snímání analogového signálu ze snímače tlaku v pneumatickém okruhu a dvou signálů ze snímače natočení volantu, jeden modul NI 9425 pro snímání polohy pedálů pokud nebyly čteny hodnoty z CAN zpráv a pro získání hodnot ze spínačů zpětných vazeb v pneumatických válcích a jeden modul NI 9472 pro spínání akčních členů systému. Poslední využívanou kartou byla PCI 8513/2, která je určena pro komunikaci CAN a je konstruována jako zásuvná karta do hlavního hardwaru

NI 3110RT. Pomocí takto sestaveného hardwaru, který je zobrazen na předchozím obrázku bylo prováděno HIL testování na prototypu vozidla.

**Software:** Používaným softwarem pro komunikaci s hardwarem je tzv. NI VeriStand, jak je popsáno v [68]. Součástí jsou různá podokna, kde uživatel nastavuje různá specifika.

**Interface:** Grafické rozhraní pro zobrazování aktuálních hodnot při HIL testování může být přes tzv. Workspace v softwaru NI VeriStand. Uživatel může využít předpřipravených tzv. Workspace Controls. Pokud je daná knihovna s prvky nedostačující může si uživatel navrhnout vlastní grafické prvky. To je i případ vývoje systému pro uzavírání diferenciálu. Takové rozhraní je programováno v prostředí softwaru NI LabVIEW. Pro základní programování v softwaru NI LabVIEW bylo vycházeno z knihy [69].

**Prototypy vozidel:** Aby bylo možné provést HIL testování, bylo nutné vyrobit prototyp vozidla, a to především mechanické části korespondující se systémem pro uzavírání diferenciálů. Jelikož tento systém byl vyvíjen i pro společnost Tatra Truck a.s., tak HIL testování probíhalo na jejich prototypových vozidlech s označením T815-7 a T158 Phoenix. Tato prototypová vozidla jsou zobrazena na následujícím obr. 10.



Obr. 10 Prototypy vozidel Tatra s označením 815-7 (vpravo) a T158 Phoenix (vlevo)

Vozidlo zobrazené vlevo je určeno pro stavební průmysl a vozidlo vpravo je určeno pro armádu nebo hasiče. Základem prototypů jsou sériová vozidla, kde došlo k úpravě a vývoji nových mechanických dílů hnacího traktu. Jednalo se o vývoj speciálních zubových spojek určených k rychlému uzavírání/otevírání diferenciálů za jízdy vozidla, s důrazem na odolnost vůči vzniku rázových momentů při jejich uzavírání a obecně koncipovaných pro přenos hnacího momentu. Tento vývoj zubových spojek a částí hnacího traktu proběhl ve společnosti Tatra Truck a.s. v rámci projektu TE 01020020, proto detaily návrhu zde nebudou popisovány. Obecně prototyp vozidla je sestaven z vyvinutých mechanických komponent tak, že zubové spojky jsou umístěny v každém diferenciálu a také pro spojení hnacích hřídelí předních náprav. To znamená, že dané prototypy s pohonem 8x8 byly z hlediska hnacího traktu sestaveny ze 4 osových diferenciálů, jednoho mezinápravového diferenciálu pro zadní nápravy a ze dvou zubových spojek pro aktivaci pohonu jednotlivých předních náprav. Všechny tyto zubové spojky jsou ovládány konzolou pneumatického válce. K němu je připojen elektroventil, který při otevření do něho vpustí tlakový vzduch z pneumatického okruhu a tím je vyvozena síla pro posuv pohyblivé části zubové spojky a dojde k jejímu spojení. K rozpojení zubové spojky dochází při uzavření elektroventilu a tím začne tlakový vzduch odcházet z pneumatického válce. To je způsobeno tím, že jeho píst je přetlačen vnitřní pružinou. Součástí pneumatického válce je koncový spínač signalizující reálné spojení zubových spojek, tedy signalizující uzavření diferenciálu nebo aktivaci pohonu předních náprav. Na následujícím obr. 11 jsou zobrazeny dílčí komponenty mechatronického systému. Zobrazení pneumatického válce je v části obrázku s označením B, kde se nachází i elektroventil. Ukázka jednoho z protikusů zubové spojky je zobrazena pod označením G.



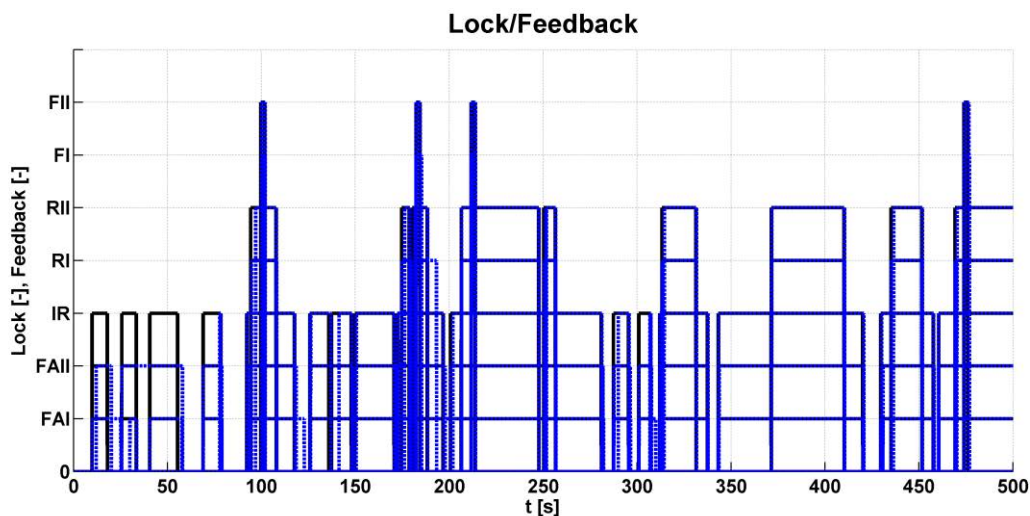
Obr. 11 Snímače a akční členy

Dále byly prototypy osazeny snímači otáček (E, F), které běžně slouží pro systém ABS. Dále je vozidlo osazeno snímačem natočení volantu a tento snímač je umístěn na jedné z pák řízení. To je zobrazeno na obrázku pod označením C. Zásobník vzduchu pneumatického okruhu je doplněn o snímač tlaku, na obrázku s označením D. Jelikož první prototyp vozidla 815-7 nedisponoval některými CAN zprávami dle normy J1939, tak musely být v kabině umístěny snímače pedálů zobrazené na obrázku s označením A pro spojkový a brzdový pedál a H pro plynový pedál. Takto byla vozidla připravena pro HIL testování a následně taky pro testování kompletního vyvíjeného systému i s prototypem ECU.

**Real-Time testování:** Pro HIL testování byly připraveny dva prototypy vozidel. Ale než se začalo testovat na nich, proběhlo ještě testování v laboratorních podmínkách. Bylo to z toho důvodu, aby zbytečně nedošlo nějakým nedopatřením k destrukci hnacího traktu vozidla při zpracování reálných signálů ze snímačů. Proto byl vytvořen testovací panel. Výstupem tohoto testu bylo odladění správného zpracovávání signálu ze snímačů a řízení akčního členu. Podrobněji jsou informace rozebrány v publikaci [71].

Z hlediska testů na prototypech vozidel byly testovány jízdní stavy na asfaltu, polních cestách a těžkém terénu. Testy obsahovaly různé způsoby jízdy, rozjezdů, zastavení, brzdění, zatáčení, testování poruch systému, testování na rovině nebo na speciálních vozovkách typu vlnovce a vozovek s definovaným sklonem. Dále byly testy prováděny v různých povětrnostních podmínkách od vysokých letních teplot přes deštivé počasí až po zasněženou a namrzlou vozovku. Součástí testování byly i jízdy po kroupené vozovce vodou pro přímý směr nebo zatáčení, dále s některými koly na asfaltu a ostatními na výrazně kluzné vozovce atd. Z hlediska HIL testování bylo cílem odladit řídicí algoritmus a otestovat mechanické, pneumatické a elektronické součásti

vyvíjeného mechatronického systému. Proto bylo snahou otestovat nejrůznější jízdní manévry s prokluzem kol a sledovat správnost fungování mechatronického systému. Ukázka dat z jednoho testu v terénu, která byla kontrolována a vyhodnocována, je na následujícím obr. 12.



Obr. 12 Průběh signálů pro akční členy (černá barva) a zpětné signály koncových spínačů (modrá barva)

Hlavními výstupními daty jsou signály *Lock* pro řízení akčních členů (černá barva) a *Feedback* - zpětné vazby z koncových spínačů pneumatických válců (modrá barva), to je zobrazeno na obr. 12. Podle grafu je jasně vidět, že docházelo k intenzivním zásahům řídicího systému, jelikož se v tomto případě jednalo o náročný blátivý terén.

Obecně všechna data byla analyzována a byla kontrolována správná funkce řídicího algoritmu. Pokud něco nebylo v pořádku, došlo k úpravě.

Součástí HIL testování bylo také velice důležité vyhodnocení času reakce na uzavření/otevření diferenciálů nebo aktivace pohonu předních náprav. Cílem bylo odladit a navrhnout celý mechatronický systém tak, aby především průměrné časy pro spojení zubových spojek byly okolo 100 ms a nižší. Tato časová reakce je důležitá z hlediska funkčnosti celého systému, aby nedocházelo k uzavření diferenciálu již při velkých rozdílných otáčkách kol. Potom by mohla nastat destrukce částí hnacího traktu. Tento problém je ošetřen i v řídicím algoritmu tím, že pokud je rozdíl otáček kol nad určený limit, tak příslušný diferenciál nesmí být již uzavřen. Přes tato opatření je důležité, aby časy reakce byly co nejmenší, protože by při uzavírání zubových spojek stále mohlo docházet k rázům, které by nevedly k destrukci částí hnacího traktu, ale byly by nepohodlné pro řidiče.

V následující tabulce jsou zobrazeny časy reakcí z odladování systému. V čase je zahrnuta doba od odeslání signálu ECU pro akční člen až po změnu zpětného signálu z koncového spínače pneumatického válce. Podrobnější popis byl publikován v časopise [72].

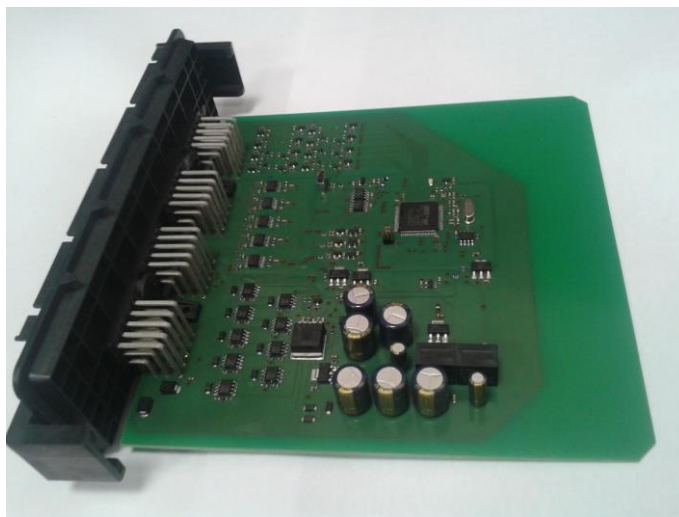
Tab. 1 Vyhodnocení času reakce na uzavření/otevření diferenciálů nebo aktivace pohonu předních náprav

	FAI	FAII	IR	RI	RII	FI	FII
Lock min. time [s]	0,01	0,01	0,02	0,01	0,01	0,07	0,07
Lock max. time [s]	0,08	1,24	1,99	1,24	0,38	2,82	1,88
Lock mean time [s]	0,07	0,19	0,22	0,14	0,09	0,77	0,35
Unlock min. time [s]	0,10	0,15	0,13	0,01	0,09	0,13	0,11
Unlock max. time [s]	1,71	4,10	3,40	1,49	2,62	2,79	2,74
Unlock mean time [s]	0,14	0,67	0,51	0,22	0,22	0,63	0,57

Přestože některé časy nebyly úplně shodné s daným požadavkem na reakci, je to stále výsledek velice uspokojující v dostatečné míře z hlediska vzniku možných momentových rázů v hnacím traktu, jelikož v kabině vozidla téměř řidič ani posádka nepoznají, že právě docházelo k uzavírání diferenciálů. Takže pozitivním výsledkem HIL testování je další odladění mechatronického systému z hlediska funkce a rázových momentů. Nejenom tyto pozitivní výsledky byly publikovány v různých časopisech, ale také jednou z dalších publikací zahrnující HIL testování tohoto systému je [70].

## 6 VÝVOJ PROTOTYPU ECU

Vývoj ECU prototypu a jeho návrh byl sestavován pomocí softwaru KiCad [73]. Jedná se o otevřenou platformu softwaru pro návrh elektronických obvodů. Součástí je rozsáhlá knihovna elektronických součástek, které uživatel používá, jak v návrhu schématu elektrického obvodu, tak při návrhu tištěného spoje. Po návrhu předběžného schématu, byly jednotlivým prvkům obvodu přiděleny pouzdra osazení. Následně návrh prototypu ECU pokračoval ve formě návrhu desky plošných spojů – DPS. Kde bylo nutností brát v úvahu základní předpoklady návrhu, kterými byly vyrobiteľnost, přesnost, vedení vodičů dle elektrických požadavků, elektromagnetická kompatibilita – EMC atd. To vše vycházelo z literatury [74], [75], [76], [77], [78] a [79] nebo z konzultací s danými odborníky na problematiku vývoje ECU. Přesto při vývoji a hledání vhodného vedení vodičů musí občas dojít ke kompromisu, aby bylo vůbec možné vše umístit a spojit na tištěném spoji. Zde například muselo být vycházeno ze zapojení vstupního konektoru zákazníka. Před samotnou výrobou prototypu ECU proběhla řada testování dílčích obvodů např. pomocí nepájivých kontaktních polí, nebo výrobou jednodušších tištěných spojů. Např. byl vytvořen vlastní mikrokontrolér pro testování periférií užitého procesoru. Pomocí něho nebo komerčního mikrokontroléru LPC1768 [80] byl vždy sestaven obvod s daným elektrickým obvodem pro jeho testování. Případně byla prováděna porovnání se simulací dílčích obvodů pomocí softwaru SIMetrix-SIMPLIS Elements [81].



*Obr. 13 Vyroběný prototyp ECU*

Po otestování dílčích elektrických obvodů bylo možné vygenerovat výrobní data prototypu ECU. To znamená, že návrh tištěného spoje byl převeden do tzv. gerber formátu a souboru se souřadnicemi a velikostmi vrtaných a frézovaných otvorů. Tyto data byla zaslána do výroby pomocí tzv. POOL servisu. Jedná se o kusovou výrobu tištěných spojů. Osazování tištěného spoje bylo provedeno manuálně, ale samozřejmě pokud by prototyp přešel do sériové výroby, bylo by prováděno strojové osazování. Potom by k základní výrobní dokumentaci přibyl ještě osazovací výkres a data obsahující souřadnice jednotlivých elektronických součástek s jejich orientací. Proto

byl připraven i plán výroby v rozsahu 50 – 100 kusů, kde by zakázka byla zadána specializované firmě na tuto výrobu, včetně dodání elektronických součástek, osazení a základního otestování. Kalkulace pro výrobu byla v rozsahu pěti poptávaných firem cca v řádech tisíců korun. Daný prototyp ECU je zobrazen na obr. 13, kde jsou vidět i dílčí části elektrického obvodu.

**Procesor:** Součástí prototypu ECU je procesor STM32417. Dle katalogového listu na stránkách [29] je založen na architektuře procesorů ARM 32-bit Cortex-M4 o frekvenci 168 MHz. Velikost flash paměti je 1 Mbyte a je jím podporována řada periférií. Z hlediska elektrického zapojení bylo vycházeno z manuálu a katalogového listu od společnosti STMicroelectronics s označením AN4488, AN2867 dostupné na stránkách [29].

**Digitální vstupy:** Elektrický obvod je sestaven z rezistorů, diody, Zenerovy diody a kondenzátoru. Vstup elektrického obvodu je spojen konektorem prototypu ECU a umožňuje připojení digitálního signálu z koncových spínačů pneumatických válců, signálů pedálů a ovládacích přepínačů a spínače. Takže pokud se jedná o připojení koncového spínače, který spíná zem, je obvod koncipován s tzv. pull-up zapojením. V případě připojení signálu z indukčních snímačů spojového a brzdového pedálu nebo kontrolních přepínačů a spínače. V tomto případě větev s pull-up rezistorem není v tištěném obvodu osazena.

**Digitální výstupy:** Z procesoru je poslán digitální výstupní signál pro aktivaci akčních členů. Jelikož se jedná o elektroventily neboli o spínanou induktivní zátěž, bylo potřeba využít dostupné elektronické prvky pro spínání takového akčního členu. Z hlediska zástavby a ručního osazování prototypu ECU byl nevhodnější tzv. Power Switch BSP752. Dále tato součástka umožňuje odesílat zpětnou vazbu o stavu z pinu ST a dle zapojení a tabulky v katalogovém listu [82] je možné např. zjistit, že je obvod rozpojen. Dále je obvod doplněn o diodu zapojenou v závěrném směru. Ta je zde umístěna z toho důvodu, že zátěž je induktivní akční člen.

**CAN:** Pro spojení s CAN sběrnici bylo nutné mezi procesor a CAN piny na konektoru ECU umístit CAN transceiver. Je to elektronické rozhraní mezi CAN protokolem a CAN sběrnici. Pro prototyp bylo použito zapojení dle katalogového listu elektronické součástky MCP2551, kde je připojen kondenzátor pro vyhlazování napájení.

**Napájení ECU:** Do konektoru prototypu ECU je připojeno napájecí napětí ze sv. 30 – stálé napětí a ze sv. 15 – napětí spínané zapalovacím klíčkem. Ze sv. 15 nebo sv. 30 jsou napájeny hlavní části jako procesor, externí paměť a plus MEMS modul. V obvodu jsou paralelně zařazeny elektrolytické kondenzátory pro vyrovnávání výkyvu napájecího napětí a dále je v obvodu DC/DC napěťový regulátor, který snižuje napětí na 9 V, aby bylo možné vést toto napětí do stabilizátorů pevného napětí pro 3,3 a 5 V. Součástí tohoto obvodu je i napěťový zdroj pro snímač natočení volantu s precizní napěťovou referencí.

**Vstupy otáček kol:** Užité indukční snímače otáček kol generují zjednodušeně řečeno sinusový signál s měnící se amplitudou a frekvencí v závislosti na otáčkách. Tento signál je zapotřebí předpřipravit procesoru do formy digitálního signálu, aby následně mohl být zpracován přes přerušeni v procesoru. Proto byla využita elektronická součástka přímo na zpracování takových signálů MAX9926 a v podstatě se jedná o komparátor, který při překročení určitých hodnot napětí generuje digitální výstup.

**Paměť:** V prototypu ECU byla využita externí paměť pro ukládání poruchových zpráv DM1 a DM2, případně nějaký omezený sběr dat. Externí paměť je typu flash a má velikost 4 Mbit. Komunikace je realizována pomocí SPI a to až do rychlosti 50 MHz. Zapojení ostatních součástek obvodu je dáno doporučením v katalogovém listu.

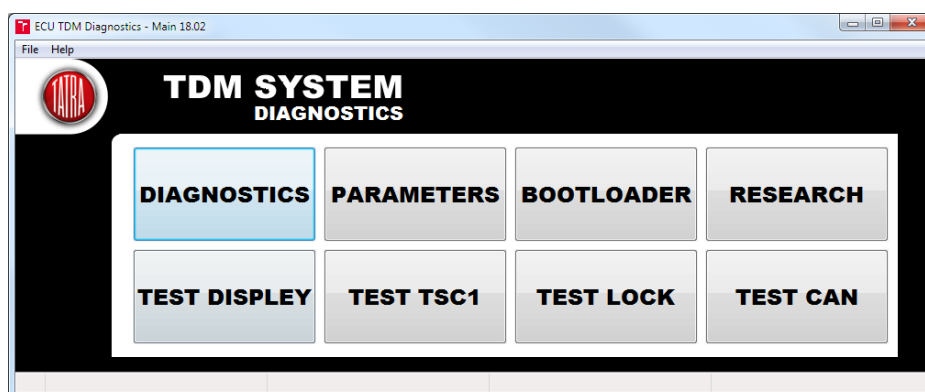
**Gyroskop:** Další nástavbou vyvíjeného systému je systém pro vyhodnocování, kde se vozidlo nachází. V tomto případě především na jakém sklonu vozovky v daném okamžiku je. Proto byl užit MEMS modul s gyroskopem a akcelerometrem. Jedná se o elektronickou součástku s názvem MPU-6000, která komunikuje pomocí SPI o přenosové rychlosti až 20 MHz. Zapojení dalších elektronických součástek obvodu vychází z katalogového listu a příslušné konfigurace.

**Analogové vstupy:** Obvody pro analogové vstupy byly sestaveny na vstupu z odporového děliče, který snižuje napětí na polovinu, to je z důvodu výstupního signálu ze snímačů natočení volantu a snímače tlaku vzduchu, který je 5 V. Procesor však má rozsah analogových vstupů jen od 0 do 3,3 V. Dále je zde užita dioda BAV99, která je zde jako ochrana proti vstupnímu napětí, které by překonalo hodnotu +3,3 nebo se snížilo pod 0 V. Následuje operační zesilovač zapojený jako sledovač napětí. Posledními součástkami jsou kondenzátory, které opět slouží k vyhlazení nebo filtraci signálu a napájení.

**USB:** Posledním významným obvodem prototypu ECU je obvod pro USB konektor, který je určen jen pro výzkumné účely a především bylo toto zapojení užito pro napájení při zprovoznování prototypu ECU. Schéma je inspirováno z prototypových desek DISCO board od STMicroelectronics.

## 7 SOFTWAREVÝ INTERFACE ECU

Pro komunikaci s prototypem ECU a nahraného programu v ní byl naprogramován softwarový interface pomocí jazyka C/C++ a knihovny wxWidget [83] určené pro programování Graphical User Interface – GUI. Do softwaru je dále implementována knihovna pro PCAN. To je hardwarové zařízení pro komunikaci s CAN sběrnici. Proto i jakákoli komunikace s ECU je realizována přes CAN zprávy. Základní grafické rozhraní softwaru je zobrazeno na obr. 14, kde uživatel má nabídku několika dalších podprogramů pro diagnostiku ECU, nastavení parametrů ECU, zavádění nového programu do procesoru. Další nabídky jsou určeny pro výzkum a vývoj. Z každého podprogramu se uživatel zase kliknutím na logo dostane do tohoto úvodního okna. Obecně v paměti procesoru je umístěn základní naprogramovaný program pro CAN komunikaci s tímto softwarem pro diagnostiku a pomocí této komunikace vše probíhá v následujících pod programech.



Obr. 14 Základní nabídka funkcí diagnostického programu

**Program Diagnostics:** První z podprogramů je určen pro diagnostiku ECU. Takže pokud ECU odesílá diagnostické CAN zprávy DM1 - Diagnostic Message 1 pro aktivní poruchy, tak při spuštění tohoto programu jsou zobrazovány. Z hlediska formátu zpráv vše vychází z normy J1939 a dle počtu zpráv jsou pro jednu poruchu odesílány ve zprávě s ID 0x18FECA09 a pokud je poruch více jsou odesílány v tzv. multipaketu s ID 0x1CCBEF09, Další možností uživatele je vyžádání zprávy DM2. To znamená, že ECU odešle dříve aktivní závady zapsané v externí paměti s ID CAN zprávy 0x18FECB09, pokud je zapsaných poruch více než jedna, jsou posílány zase v tzv. multipaketu ve stejném formátu jako DM1. Tlačítko softwaru DM3 odešle požadavek na vymazání dříve aktivních závad.

**Program Parameters:** Program umožňuje nastavovat různé parametry pro řídicí algoritmus. Prvně je nastavován typ vozidla, pro který je mechatronický systém určen, takže např. vozidlo s pohonem 4x4 až 10x10. Dále je nastavován počet zubů kola pro ABS senzory, obvod



pneumatiky přední a zadní nápravy. Další nastavitelnou sekcí je tzv. Vehicle Dimensions, kde jsou nastavovány rozvory a rozchody náprav a poloměr rejdu v rovině středu kola. Následuje nastavení, zda některé signály chce uživatel získávat pro řídicí algoritmus z dalších ECU. Pokud prototyp ECU disponuje nastavením pro sledování okolí, je zpřístupněno nastavení polohy MEMS modulu, tedy v podstatě poloha řídicí jednotky ve vozidle vůči základnímu souřadnému systému vozidla dle ISO. Následují nastavitelné hodnoty senzoru tlaku vzduchu v pneumatickém okruhu a tabulka nastavení snímače natočení volantu vzhledem k zatočení pravého a levého kola přední nápravy. Průběh přenesení nových parametrů do ECU nebo naopak vyčtení aktuálních parametrů je pomocí tlačítek (LOAD, SAVE). Po těchto operacích je nutné kliknout na tlačítko RESET pro restartování ECU.

**Program Bootloader:** Důležitým podprogramem je tzv. Bootloader, který umožňuje otevřít (OPEN) zkompileovaný kód programu s řídicím algoritmem ve formátu hex. Zároveň ho zase ze svoji paměti může smazat pomocí tlačítka CLEAR. Pokud je soubor načten a uživatel zmáčkne tlačítko UPLOAD, software nejdříve naváže spojení s ECU. Pokud je vše v pořádku, tak začne odesílat CAN zprávy obsahující postupně celý nový zaváděný program s řídicím algoritmem do paměti procesoru. Výhodou tohoto rozhraní je, že při každé změně řídicího algoritmu nemusí dodavatel jet k zákazníkovi, aby mu tam nahrál novou verzi programu, ale stačí to poslat elektronicky a uživatel pomocí tohoto softwaru a běžně používaného PCAN rozhraní může nahrát program samostatně.

**Program Research:** Tento podprogram, je určen pro vývoj a výzkum tohoto systému. Tento program funguje jako online zobrazování dat a stavů vstupních, výstupních a i vnitřních veličin řídicího algoritmu. Zároveň je zde možnost sběru dat v rozsahu posledních 3 minut. Ty jsou následně zpracovávány a vyhodnocena.

**Program Test displeje:** Jedná se o program pro testování a ověření komunikace dotykového displeje. Tento program byl využit pro vývojové práce na dotykovém displeji, aniž by musel být k dispozici prototyp ECU.

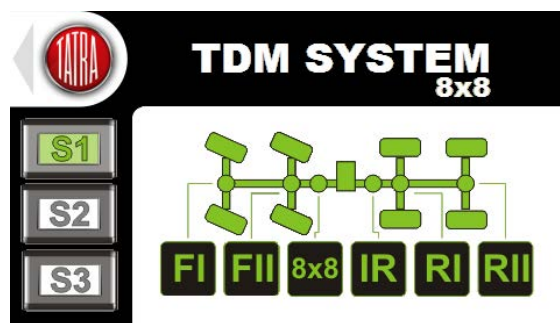
**Program Test TSC1:** Další program byl naprogramován pro účely testování reakce motoru na zasílané omezení pomocí CAN zprávy TSC1. V grafickém rozhraní uživatel nastaví příslušnou konfiguraci a po stisknutí tlačítka SEND se začne cyklicky dle nastavení odesílat CAN zpráva TSC1. Na ni podle požadavku a dalších priorit v řídicí jednotce motoru, začne motor reagovat. To vše pokud tuto zprávu a zdrojovou adresu v ID zprávy podporuje.

**Program Test lock:** Tento program je ve fázi vývoje a jedná se o program určený k manuálnímu testování akčních členů.

**Program Test CAN:** Poslední program, je určen pro odesílání uživatelsky definovaných CAN zpráv pro různá testování. Uživatel může odesílat jednu zprávu nebo celý paket.

## 8 DISPLEJ

Součástí celého vyvíjeného systému je i dotykový displej, který v první fázi byl užit s označením 32F746GDISCOVERY. Nyní je vývoj ve fázi užití i displeje Opus 3Ae.



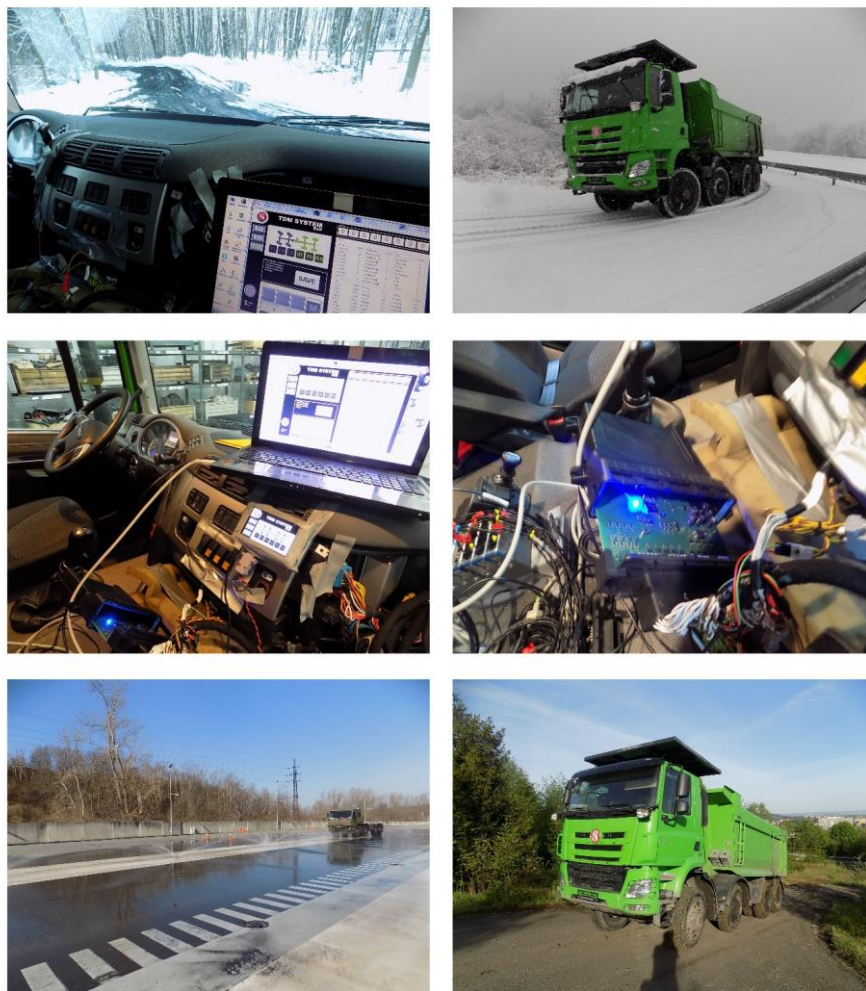
Obr. 15 Dotykový displej pro vyvíjený mechatronický systém

**Hardware:** Z hlediska hardwaru byl tedy užít dotykový displej 32F746GDISCOVERY, který je založen na procesoru STM32F746G s pamětí 1 Mbyte. Tento hardware disponuje různými periferiemi pro komunikaci.

**Software:** Vnitřní program byl programován v jazyce C++ ve vývojovém prostředí mbed [80]. Kde v první fázi byla navržena grafika a obrázky byly konvertovány do matice a umístěny do souborů projektu. S nimi již pracuje naprogramovaný algoritmus a dle přijímaných CAN zpráv a dat z nich zobrazuje příslušné obrázky nebo texty na obrazovce displeje. Tímto prvkem systému je řidič informován o daném stavu a může nastavovat příslušné jízdní módy ECU.

## 9 FINÁLNÍ TESTOVÁNÍ MECHATRONICKÉHO SYSTÉMU

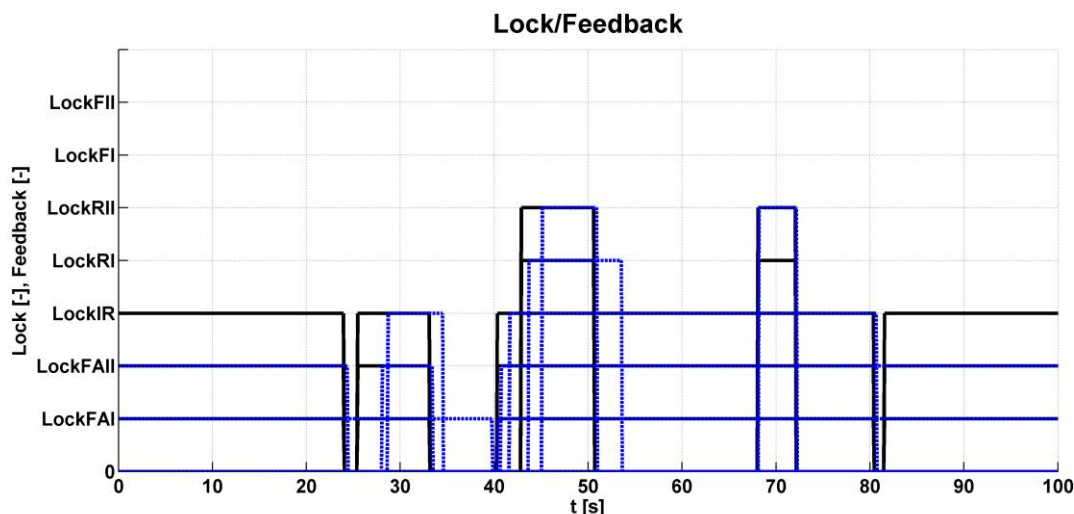
Celkové testování vyvíjeného autonomního mechatronického systému pro uzavírání diferencíálů probíhalo opět na prototypu vozidla s pohonem 8x8. Systém byl sestaven již z finálních senzorů, akčních členů, prototypu ECU, diagnostického softwaru a dotykového displeje. Vše je názorně zobrazeno na obr. 16. Testování probíhalo znovu v různých podmínkách od asfaltové silnice po nejnáročnější terény a povětrnostní podmínky.



*Obr. 16 Testování kompletního vyvinutého autonomního mechatronického systému pro uzavírání diferencíálů*

Během testování již byly laděny jen drobné úpravy a nebyl předpokládán zásadní zásah do filozofie celého systému. Docházelo jen k ladění řídicích parametrů, aby systém reagoval rychle a spolehlivě ve všech situacích. Na následujícím grafu je zobrazena ukázka sbíraných dat během

jednoho z mnoha testů. Zde je jízda lehčím terénem okolo rychlosti vozidla 10 km/h. V grafu je průběh signálů pro aktivaci/deaktivaci akčních členů a signály koncových spínačů pneumatických válců, zobrazen na obr. 17. Signály pro akční členy jsou zobrazeny černou barvou a pro zpětné vazby modře. Zde je vidět, že během testování byl téměř po celou dobu aktivován pohon všech kol a v některých situacích byly uzavřeny zadní osově diferenciály. Pokud podmínky pro uzavření pominuly, diferenciály byly otevřeny. I z grafu vyplývá, že terén nebyl jeden z nejnáročnějších, jelikož nedocházelo k uzavírání předních osových diferenciálů.



Obr. 17 Průběh signálů pro akční členy (černá barva) a zpětné signály koncových spínačů (modrá barva)

Jak již bylo několikrát zmíněno i pro toto ověření kompletního navrženého mechatronického systému bylo provedeno velké množství testů, proto je zde zobrazena a jen jednoduše popsána ukázka jednoho z nich.

## ZÁVĚR

Tato práce se zabývala vývojem autonomního mechatronického systému pro uzavírání diferenciálů. Ten je uzpůsoben pro různé typy vozidel od osobních přes nákladní až po různé speciály a traktory. V jednotlivých kapitolách teze byl velice stručně popsán vývoj a testování dílčích částí mechatronického systému. Tento systém je sestaven z řídicího algoritmu, prototypu ECU, senzorů, a akčních členů. Základním principem je snímání otáček kol, z nichž je vyhodnocován skluz. Pokud řídicí algoritmus v ECU vyhodnotí aktuální situaci tak, že dochází k prokluzu při jízdě vozidla, udělá příslušné kroky k aktivaci pohonu všech kol. Pokud ani tento zásah nestačí, postupně uzavírá zadní osově diferenciály a následně přední osově diferenciály. Aby bylo toto možné realizovat, je akční člen sestaven z elektroventilu, který se po aktivaci otevře a tím pustí tlakový vzduch do pneumatického válce s konzolou. Tím je vyvozena síla, která spojuje speciálně navržené zubové spojky a tím dochází ke spojování hnacích hřídelů předních náprav (aktivace pohonu všech kol) nebo uzavření příslušného diferenciálu.

V první fázi vývoje byl navrhován řídicí algoritmus. Následovala první z metod ověřování řídicího algoritmu tzv. MIL testování. Dle odhadu během tohoto vývoje a v této fázi testování bylo nalezeno a provedeno asi zhruba 70 % chyb a úprav, proto byl tento postup testování velice významný při vývoji. Druhou metodou nejenom pro ověření řídicího algoritmu, ale i pro ověření senzorů a akčních členů prototypu vozidla bylo použito tzv. HIL testování. V této fázi testování bylo dle odhadu odhaleno 28 % chyb z celkového počtu. Snahou potom bylo celý systém naladit tak, aby se časy reakce systému pohybovaly pod 0,1 s. Paralelně probíhal vývoj prototypu ECU. Dle požadovaných vstupů a výstupů byly navrženy elektronické obvody pro zpracování a

předávání signálů pro procesor. Pro komunikaci s ECU byl naprogramován grafický interface. Dále byly naprogramovány podprogramy pro nastavování uživatelských parametrů ECU, diagnostiku ECU a testování dílčích částí. Aby byl řidič informován o stavu aktivace pohonu všech kol nebo uzavření příslušných diferenciálů, byl naprogramován a popsán v této práci dotykový displej, který tuto funkci umožňuje. Poslední z hlavních kapitol popisuje ukázkou finálního testování kompletního vyvinutého mechatronického systému odpovídající verzi pro sériovou výrobu. V této fázi lze odhadnout nalezení 2 % chyb nebo spíše to obsahuje jenom doladování parametrů řízení.

Obecným shrnutím této práce je, že byl vyvinut autonomní mechatronický systém pro uzavírání diferenciálů včetně jeho otestování na prototypu vozidla krom návrhu speciálních zubových spojek. Z testů a reakcí zákazníka vyplývá, že systém funguje správně a je přínosem pro vozidla, aby nedocházelo k jejich uvíznutí v nepříznivých terénních podmínkách a zároveň byl chráněn hnací trakt před neodborným používáním uzavírání diferenciálů nezkušenými řidiči. Jak již bylo zmíněno, v rámci projektu TE 01020020 a ve spolupráci se společností Tatra Truck a.s. byl samotný řídicí algoritmus i uzpůsoben pro jejich vozidla, kde ho implementovali do jejich řídicí jednotky. V rámci obecného výzkumu tohoto mechatronického systému bylo navrženo vlastní řešení ECU, modulárního řídicího algoritmu, softwaru pro komunikaci s ECU a provedeno testování popsané v této práci.

Pokračováním vývoje bude implementace na další typy vozidel a spolupráce s dalšími zákazníky tohoto systému. Předpokladem je rozšiřování nadstavby pro predikci okolí vozidla, aby byl lépe využit hnací moment. Dále se rozvíjí metoda pro testování ECU pomocí HIL tak, že pomocí výpočtových modelů vozidla jsou generovány signály nahrazující reálné snímače a tím sledována reakce prototypu ECU s řídicím algoritmem. Z prototypu ECU jsou odesílány signály akčním členům ve formě výpočtového modelu a tím je možné v reálném čase testovat tento systém a prototyp ECU se zásahy reálného řidiče bez nároků na výrobu prototypu vozidla. Dalším předpokládaným vývojem bude minimalizace rozměrů prototypu ECU. Posledním předpokládaným dalším postupem vývoje bude využití tzv. neuronových sítí, aby se řídicí algoritmus mohl během užívání dále učit a přizpůsobovat vozidlu a jízdním situacím.

## POUŽITÉ INFORMAČNÍ ZDROJE

- [1] KERNIGHAN, Brian W a Dennis M RITCHIE. *Programovací jazyk C*. 1. Brno: Computer Press, 2006. ISBN 80-251-0897-X.
- [2] LIBERTY, Jesse. *Naučte se C za 21 dní*. 2., aktualiz. vyd. Brno: Computer Press, 2007. ISBN 978-80-251-1583-1.
- [3] Simulink. In: *MathWorks* [online]. United States: MathWorks, 2018 [cit. 2018-09-25]. Dostupné z: <https://www.mathworks.com/products/simulink.html>
- [4] *ARM Keil* [online]. Cambridge, UK: Arm Limited, 2018 [cit. 2018-09-25]. Dostupné z: <http://www.keil.com/>
- [5] IAR Embedded Workbench. In: *IAR Embedded Workbench* [online]. Uppsala, Sweden: IAR Systems, 2018 [cit. 2018-09-25]. Dostupné z: <https://www.iar.com/iar-embedded-workbench/>
- [6] *AdaCore* [online]. New York, USA: AdaCore, 2018 [cit. 2018-09-25]. Dostupné z: <https://www.adacore.com/>
- [7] Allegro PCB Designer. *Www.cadence.com* [online]. Cadence Design Systems, 2018 [cit. 2018-09-25]. Dostupné z: [https://www.cadence.com/content/cadence-www/global/en\\_US/home/tools/pcb-design-and-analysis/pcb-layout/allegro-pcb-designer.html](https://www.cadence.com/content/cadence-www/global/en_US/home/tools/pcb-design-and-analysis/pcb-layout/allegro-pcb-designer.html)
- [8] *EasyEDA* [online]. China: EasyEDA, 2018 [cit. 2018-09-25]. Dostupné z: <https://easyeda.com/>
- [9] EAGLE. In: *Autodesk* [online]. Kalifornie, USA: Autodesk, 2018 [cit. 2018-09-25]. Dostupné z: <https://www.autodesk.com/products/eagle/overview>

- [10] OrCAD PCB Designer. In: *OrCAD* [online]. Hillsboro, Oregon: OrCAD, 2018 [cit. 2018-09-25]. Dostupné z: <https://www.orcad.com/products/orcad-pcb-designer/overview>
- [11] Multisim. In: *National Instruments* [online]. Austin, Texas: National Instruments, 2018 [cit. 2018-09-25]. Dostupné z: <http://www.ni.com/cs-cz/shop/electronic-test-instrumentation/application-software-for-electronic-test-and-instrumentation-category/what-is-multisim.html>
- [12] FlexECU. In: *Mathworks* [online]. United States: The MathWorks, 2018 [cit. 2018-09-25]. Dostupné z: [https://www.mathworks.com/products/connections/product\\_detail/flexecu.html](https://www.mathworks.com/products/connections/product_detail/flexecu.html)
- [13] PRODUCTS: Discover OpenECU. In: *Pi-Innovo* [online]. Plymouth, MI 48170-3765 United States of America: Pi Innovo, 2018 [cit. 2018-09-25]. Dostupné z: [www.pi-innovo.com](http://www.pi-innovo.com)
- [14] Bosch Motorsport - Overview. In: *Bosch Motorsport* [online]. Německo: Bosch Engineering, 2018 [cit. 2018-09-25]. Dostupné z: <http://www.bosch-motorsport.de/content/downloads/Products/9007213411107467.html>
- [15] Rapid development, RT simulace, modelování. In: *REX Controls* [online]. Plzeň: REX Controls, 2018 [cit. 2018-09-25]. Dostupné z: <https://www.rexcontrols.cz/rapid-development-rt-simulace-modelovani>
- [16] *dSpace* [online]. Paderborn Germany: dSpace, 2018 [cit. 2018-09-25]. Dostupné z: <https://www.dspace.com/en/pub/home/products/systems/ecutest.cfm>
- [17] NovaCarts HiL Simulation. In: *MicroNova* [online]. Vierkirchen: MicroNova, 2018 [cit. 2018-09-25]. Dostupné z: <https://www.micronova.de/en/testing/novacarts-hil-simulation.html>
- [18] Modular Test Hardware: VT System. In: *Vector* [online]. Stuttgart, Germany: Vector, 2018 [cit. 2018-09-25]. Dostupné z: [https://vector.com/vi\\_vt-system\\_en.html](https://vector.com/vi_vt-system_en.html)
- [19] Hardware-in-the-Loop Testing (HIL) for Real-Time Plant Simulation. In: *Speedgoat* [online]. Liebefeld Switzerland: Speedgoat, 2018 [cit. 2018-09-25]. Dostupné z: [https://www.speedgoat.com/applications-industries/applications/plant-simulation-hil?gclid=EAIaIQobChMI7uimsjV3QIV1eJ3Ch2G4AN6EAAAYASAAEgJCqPD\\_BwE](https://www.speedgoat.com/applications-industries/applications/plant-simulation-hil?gclid=EAIaIQobChMI7uimsjV3QIV1eJ3Ch2G4AN6EAAAYASAAEgJCqPD_BwE)
- [20] Hardware-in-the-Loop (HIL) Simulation. In: *National Instruments* [online]. Austin, Texas: National Instruments, 2018 [cit. 2018-09-25]. Dostupné z: <http://www.ni.com/cs-cz/innovations/automotive/hardware-in-the-loop.html>
- [21] Transportation. In: *US Micro Products* [online]. Austin: US Micro Products, 2016 [cit. 2018-09-25]. Dostupné z: <http://www.usmicroproducts.com/applications/transportation>
- [22] OPUS Control Panels. In: *Topcon* [online]. Geisenheim, Germany: Topcon Electronics, 2018 [cit. 2018-09-25]. Dostupné z: <https://topcon-electronics.de/en/products/overview.html>
- [23] Continental's 3D Touch Surface Display Receives Highest Honor at CES 2018 Innovation Awards. In: *Continental AG* [online]. Hannover, Germany: Continental, 2018 [cit. 2018-09-25]. Dostupné z: <https://www.continental-corporation.com/en/press/press-releases/2018-01-04-3d-touch-display-118076>
- [24] INCA Base Product. In: *ETAS* [online]. Stuttgart Germany: ETAS, 2018 [cit. 2018-09-25]. Dostupné z: <https://www.etas.com/en/products/inca.php>
- [25] Standards. In: *ASAM* [online]. Hoehenkirchen, Germany: ASAM e.V., 2018 [cit. 2018-09-25]. Dostupné z: <https://www.asam.net/standards/>
- [26] STELZENEDER, H., Franz X. a H. AITZETMÜLLER. ADM A NEW DRIVE-TRAIN MANAGEMENT. In: *FISITA world automotive congress; Seoul 2000 FISITA world automotive congress, automotive innovation for the new millennium*. Korean: Korean Society of Automotive Engineers, 2000, s. 1-7. ISBN 8985000004.
- [27] ADM 2. In: *Products for Trucks* [online]. Friedrichshafen Germany: ZF Friedrichshafen, 2018 [cit. 2018-09-11]. Dostupné z: [https://www.zf.com/products/en/trucks/products\\_29087.html](https://www.zf.com/products/en/trucks/products_29087.html)
- [28] *Meritor* [online]. United States: Meritor, 2018 [cit. 2018-09-25]. Dostupné z: <https://www.meritor.com/>

- [29] STM32F417VG. In: *STMicroelectronics* [online]. Geneva, Switzerland: STMicroelectronics, 2018 [cit. 2018-09-11]. Dostupné z: <https://www.st.com/en/microcontrollers/stm32f417vg.html>
- [30] STM32CubeMX. In: *STMicroelectronics* [online]. Geneva, Switzerland: STMicroelectronics, 2018 [cit. 2018-09-11]. Dostupné z: <https://www.st.com/en/development-tools/stm32cubemx.html>
- [31] TrueSTUDIO. In: *Atollic* [online]. Jönköping, Sweden: Atollic, 2018 [cit. 2018-09-11]. Dostupné z: <https://atollic.com/truestudio/>
- [32] STM32CubeF4. In: *STMicroelectronics* [online]. Geneva, Switzerland: STMicroelectronics, 2018 [cit. 2018-09-11]. Dostupné z: <https://www.st.com/en/embedded-software/stm32cubef4.html>
- [33] *J1939 - Recommended Practice for a Serial Control and Communications Vehicle Network*. 1. United States: SAE International, 2005.
- [34] *J1939-21 - DATA LINK LAYER*. 1. United States: SAE International, 2001.
- [35] *J1939-31 - Network Layer*. 1. United States: SAE International, 2004.
- [36] *J1939-71 - Vehicle Application Layer*. 1. United States: SAE International, 2008.
- [37] *J1939-73 - Application Layer - Diagnostics*. 1. United States: SAE International, 2006.
- [38] *J1939-81 - NETWORK MANAGEMENT*. 1. United States: SAE International, 2003.
- [39] ŠTĚRBA, Pavel. *Elektronika a elektrotechnika motorových vozidel: Seřizování, diagnostika závad a chybové kódy OBD*. Brno: CPRESS, 2013. ISBN 978-80-264-0271-8.
- [40] BAUER, František. *Traktory a jejich využití*. 2. vyd. Praha: Profi Press, 2013. ISBN 978-80-86726-52-6.
- [41] SELECKÝ, Matuš. *Arduino: uživatelská příručka*. Brno: Computer Press, 2016. ISBN 978-80-251-4840-2.
- [42] VÁŇA, Vladimír. *ARM pro začátečníky*. 1. Praha: BEN - technická literatura, 2009. ISBN 978-80-7300-246-6.
- [43] IMU Data Fusing: Complementary, Kalman, and Mahony Filter. In: *OlliW's Bastelseiten* [online]. Germany: OlliW's Bastelseiten, 2016, 2013 [cit. 2018-09-11]. Dostupné z: <http://www.olliw.eu/2013/imu-data-fusing/#chapter42>
- [44] ST-LINK/V2. *STMicroelectronics* [online]. Geneva, Switzerland: STMicroelectronics, 2018 [cit. 2018-09-11]. Dostupné z: <https://www.st.com/en/development-tools/st-link-v2.html>
- [45] J-Link Debug Probes. In: *SEGGER Microcontroller GmbH* [online]. Monheim am Rhein, Germany: SEGGER Microcontroller, 2018 [cit. 2018-09-11]. Dostupné z: <https://www.segger.com/products/debug-probes/j-link/>
- [46] PCAN-USB: CAN Interface for USB. In: *PEAK-System Technik GmbH* [online]. Darmstadt, Germany: PEAK-System Technik, 2018 [cit. 2018-09-11]. Dostupné z: <https://www.peak-system.com/PCAN-USB.199.0.html?L=1>
- [47] KUČERA, Pavel. *Mechatronický přístup v dynamice vozidel*. Brno, 2015. Dizertační práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství. Vedoucí práce Prof. Ing. Václav Píštěk, DrSc.
- [48] KUČERA, P.; PÍŠTĚK, V. A COMPUTATIONAL ENGINE MODEL IN SIMULINK SOFTWARE. In KOKA 2016. Brno: 2016. s. 109-112. ISBN: 978-80-214-5379- 1.
- [49] KUČERA, P.; PÍŠTĚK, V. A Computational Model of ATV Vehicle for Real- Time Simulation. In Engineering Mechanics 2015. Applied Mechanics and Materials. 2016. s. 242-247. ISBN: 978-3-03835-700- 1. ISSN: 1662- 7482.
- [50] KUČERA, P.; PÍŠTĚK, V. PODÉLNÁ A PŘÍČNÁ DYNAMIKA NÁKLADNÍHO VOZIDLA S POHONEM 6x6. *Perners' Contacts*, 2015, roč. X, č. 3, s. 81-86. ISSN: 1801-674X.
- [51] KUČERA, P.; PÍŠTĚK, V. Longitudinal and lateral dynamics of a commercial vehicle in Simulink software. In *Proceeding of International Conference Transport Means 2015*.

- Transport Means. Kaunas Univ Technol, Kaunas, LITHUANIA: KAUNAS UNIV TECHNOLOGY PRESS, K DONELAICIO 73, KAUNAS LT 3006, LITHUANIA, 2015. s. 458-461. ISBN: 9955-09-935- 6. ISSN: 1822-296X.
- [52] KUČERA, P.; PÍŠTĚK, V. Virtual prototype of a heavy duty off- road truck driveline in Simulink software. In *Proceeding of International Conference Transport Means 2014. Transport Means*. K. Donelaičio st. 73, LT- 44029 Kaunas: Kaunas University of Technology, 2014. s. 5-8. ISBN: 9955-09-935- 6. ISSN: 1822- 296X.
- [53] KUČERA, P.; PÍŠTĚK, V. A Computational Model of Powertrain Components in Simulink. In *Engineering Mechanics 2014. Engineering mechanics 2014*. 1. Brno University of Technology, 2014. s. 344-347. ISBN: 978-80-214-4871- 1. ISSN: 1805- 8248.
- [54] KUČERA, P.; PÍŠTĚK, V. SIMULINK BLOCK LIBRARY FOR ASSEMBLING A VEHICLE. *Perners' Contacts*, 2014, roč. IX, č. 2, s. 38-47. ISSN: 1801- 674X.
- [55] KUČERA, P.; PÍŠTĚK, V. VIRTUAL DIESEL ENGINE IN SIMULINK. *Perners' Contacts*, 2013, roč. VIII, č. 2, s. 95-105. ISSN: 1801- 674X.
- [56] KUČERA, P.; PÍŠTĚK, V. TRANSMISSION COMPUTATIONAL MODEL IN SIMULINK. *Perners' Contacts*, 2013, roč. VIII, č. 4, s. 37-48. ISSN: 1801- 674X
- [57] KARNOPP, Dean. Computer Simulation of Stick-Slip Friction in Mechanical Dynamic Systems. *Journal of Dynamic Systems, Measurement, and Control*. 1985, 107(1). DOI: 10.1115/1.3140698. ISSN 00220434. Dostupné také z: <http://DynamicSystems.asmedigitalcollection.asme.org/article.aspx?articleid=1403660>
- [58] PACEJKA, Hans B. *Tyre and Vehicle Dynamics*. 2. vydání. Oxford: Butterworth-Heinemann, 2006. ISBN 0-7506-6918-7.
- [59] KUČERA, P.; PÍŠTĚK, V. Mechatronic System of Automatic and Manual Differential Lock Control - Vehicle Turnig. In Proceedings of the 2016 17th International Conference on Mechatronics – Mechatronika (ME) 2016. 1st edition. Prague: Czech Technical University in Prague, 2016. s. 307-310. ISBN: 978-80-01-05882- 4.
- [60] KUČERA, P.; PÍŠTĚK, V. Automatic and Manual Differential Lock Control in a Truck – Cornering. In 20th International Conference Transport Means 2016 Proceedings Part I. Transport Means. Kaunas University of Technology, Kaunas, LITHUANIA: KAUNAS UNIV TECHNOLOGY PRESS, K DONELAICIO 73, KAUNAS LT 3006, LITHUANIA, 2016. s. 98-101. ISSN: 1822-296X.
- [61] KUČERA, P.; PÍŠTĚK, V.; PORTEŠ, P. AUTOMATIC DIFFERENTIAL LOCK CONTROL IN A TRUCK – STRAIGHT DRIVE. In *Engineering Mechanics 2016*. 1. Prague: Institute of Thermomechanics Academy of Sciences of the Czech Republic, v.v.i., Prague, 2016. s. 334-337. ISBN: 978-80-87012-59- 8
- [62] KUČERA, P.; PÍŠTĚK, V.; PORTEŠ, P. AUTOMATICKÉ ŘÍZENÍ UZAVÍRÁNÍ DIFERENCIÁLU NÁKLADNÍHO VOZIDLA PŘI JÍZDĚ PŘÍMÝM SMĚREM. *Perners' Contacts*, 2016, roč. XI, č. 1, s. 115-121. ISSN: 1801-674X.
- [63] DABNEY, James a Thomas L HARMAN. *Mastering Simulink*. Upper Saddle River: Pearson Prentice Hall, 2004, 376 s. ISBN 0-13-142477-7.
- [64] GREPL, Robert. *Kinematika a dynamika mechatronických systémů*. Brno: CERM - Akademické nakladatelství, 2007. ISBN 978-80214-3530-8.
- [65] BUDYNAS, Richard G. a J. Keith. NISBETT. *Shigley's mechanical engineering design*. Tenth edition. New York: McGraw-Hill Education, 2015. ISBN 00-733-9820-9.
- [66] Adams/Tire help - MSC Adams 2011. In: *MSC Software* [online]. Newport Beach, CA: MSC Software, 2018, 2014 [cit. 2018-09-11]. Dostupné z: <https://simcompanion.mscsoftware.com/infocenter/index?page=content&id=DOC9837&cat=1VMO50&actp=LIST>
- [67] *National Instruments* [online]. Austin, Texas: National Instruments, 2018 [cit. 2018-09-11]. Dostupné z: <http://www.ni.com/cs-cz.html>

- [68] VeriStand. In: *National Instruments* [online]. Austin, Texas: National Instruments, 2018 [cit. 2018-09-11]. Dostupné z: <http://www.ni.com/veristand/>
- [69] VLACH, Jaroslav, Josef HAVLÍČEK a Martin VLACH. *Začínáme s LabVIEW*. Praha: BEN - technická literatura, 2008. ISBN 978-80-7300-245-9.
- [70] PORTEŠ, P.; KUČERA, P.; PÍŠTĚK, V.; FOJTÁŠEK, J.; ZHÁŇAL, L. MODERN TOOLS FOR VEHICLE DEVELOPMENT. In *Engineering Mechanics 2017*. 1. 2017. s. 54-57. ISBN: 978-80-214-5497-2.
- [71] KUČERA, P.; PÍŠTĚK, V. Using HIL testing for Differential Lock Control. In *Transport Means 2017 Proceedings. Transport Means*. 1. Kaunas University of Technology, Kaunas, LITHUANIA: 2017. s. 640-643. ISSN: 1822-296X.
- [72] KUČERA, P.; PÍŠTĚK, V. Testing of the Mechatronic Robotic System of the Differential Lock Control on a Truck. *International Journal of Advanced Robotic Systems*, 2017, roč. 14, č. 5, s. 1-7. ISSN: 1729-8814.
- [73] *KiCad EDA* [online]. Vesoul, France: KiCad EDA, 2018 [cit. 2018-09-11]. Dostupné z: <http://kicad-pcb.org/>
- [74] JAN, Zdeněk, Bronislav ŽDÁNSKÝ a Jindřich KUBÁT. *Automobily. 5, Elektrotechnika motorových vozidel I*. 3. vydání. Brno: Avid, 2012. ISBN 978-80-87143-22-3.
- [75] JAN, Zdeněk, Jindřich KUBÁT a Bronislav ŽDÁNSKÝ. *Automobily. 6, Elektrotechnika motorových vozidel I*. 3. Brno: Avid, 2013. ISBN 978-80-87143-27-8.
- [76] ZÁHLAVA, Vít. *Metodika návrhu plošných spojů*. 1. Praha: Vydavatelství ČVUT, 2000. ISBN 80-010-2193-9.
- [77] ŠANDERA, Josef. *Návrh plošných spojů pro povrchovou montáž*. 1. Praha: BEN - technická literatura, 2006. ISBN 80-730-0181-0.
- [78] DOLEČEK, Jaroslav. *Moderní učebnice elektroniky - 1. díl Základy elektroniky, ideální a reálné prvky: rezistor, kondenzátor, cívka*. 1. Praha: BEN - technická literatura, 2005. ISBN 80-730-0146-2.
- [79] DOLEČEK, Jaroslav. *Moderní učebnice elektroniky - 2. díl Polovodičové prvky a elektronky*. 1. Praha: BEN - technická literatura, 2005. ISBN 80-730-0161-6.
- [80] Mbed LPC1768. In: *Arm MBED* [online]. Cambridge, UK: Arm Limited, 2018 [cit. 2018-09-11]. Dostupné z: <https://os.mbed.com/platforms/mbed-LPC1768/>
- [81] SIMetrix. In: *SIMetrix Technologies Ltd* [online]. Berkshire, United Kingdom: SIMetrix Technologies, 2018 [cit. 2018-09-11]. Dostupné z: <https://www.simetrix.co.uk/products/simetrix.html>
- [82] BSP752R. *Infineon Technologies AG* [online]. Munich, Germany: Infineon Technologies, 2018 [cit. 2018-09-11]. Dostupné z: <https://www.infineon.com/cms/en/product/power/smart-low-side-high-side-switches/automotive-smart-high-side-switch-profet/bsp752r/>
- [83] *WxWidgets* [online]. United kingdom: wxWidgets, 2018 [cit. 2018-09-11]. Dostupné z: <https://www.wxwidgets.org/>



## **ABSTRACT**

This thesis deals with the development of an autonomous mechatronic system for locking of vehicle differentials. This system is designed to prevent the vehicle from being jammed under adverse adhesion conditions. The thesis describes the development of the control algorithm, the electronic control unit – ECU and the diagnostic and the implementation software. In addition, the Model in the Loop - MIL testing of the control algorithm, Hardware in the Loop - HIL testing on the prototype vehicle and the overall testing of the mechatronic system for differential locking are described. Last but not least, the programming and implementation of the touch screen for communication with the electronic control unit is described.