ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA ELEKTROTECHNICKÁ

DIPLOMOVÁ PRÁCE

Diagnostický systém pro závodní automobil CTU Cartech FS02

2011

Autor: Bc. Jakub Šalda

Prohlášení

Prohlašuji, že jsem svou diplomovou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v přiloženém seznamu.

V Praze dne:_____ Podpis:_____

Poděkování

Rád bych poděkoval lidem, bez jejichž podpory bych tuto práci nemohl realizovat. Jsou jimi vedoucí elektrotechnické skupiny CTU CarTech Ing. Lukáš Čarek, vedoucí mé práce Ing. Pavel Burget, Ph.D., členové týmu CTU CarTech a své rodině za podporu v průběhu celého studia.

Abstrakt

Cílem této diplomové práce bylo navrhnout a realizovat jednotku pro záznam dat z formulového vozu FS02 vyvíjeným týmem CarTech z Českého vysokého učení technického pro soutěž Formula student / SAE Team. V práci je popsáno hardwarové zapojení jednotky s procesorem PowerPC, ARM7 a implementace softwaru. Deska je propojena s vozem pomocí sběrnice CAN, z níž získává informace, které potom zapíše na záznamové zařízení. V práci je dále popsán postup jak pracovat s navrženou deskou a jak jí používat.

Klíčová slova

CAN, LPC21xx, OMK, MPC5200

Abstract

The aim of this thesis was to design and implement the record data unit from the car FS02 developt by CarTech team from the Czech Technical University for competition Formula Student / SAE Team. The thesis describes the involvement of the hardware with PowerPC, ARM7 and software implementations. The board is connected to the car by CAN bus, from which it receives information, which then writes to a recording device. The thesis also describes how to work with the proposed board and how to use it.

Keywords

CAN, LPC21xx, OMK, MPC5200

Katedra elektromagnetického pole ***Školní rok:*** 2009/2010

ZADÁNÍ DIPLOMOVÉ PRÁCE

Student: **Bc. Jakub Šalda**

Obor: Telekomunikace a radiotechnika

Název tématu: Diagnostický systém pro závodní automobil CTU Cartech FS02
Diagnostic Systém for the Race Car CTU CarTech FS02

Zásady pro vypracování:

1. Seznamte se s komerčně dodávanými systémy pro diagnostiku závodních automobilů.
2. Proveďte rešerši v oblasti senzorů používaných pro diagnostiku automobilů.
3. Vyberte vhodné senzory a navrhněte jejich připojení k jednotce.
4. Seznamte se s procesorovým modulem PPC5200.
5. Rozhodněte o použití samostatné nosné desky procesorového modulu nebo využití senzorického modulu s procesorem LPC21xx.
6. Navrhněte hardware pro modul PPC5200 (případně hardware s procesorem LPC21xx).
7. Implementujte vhodný software, který umožní demonstraci funkce a následné využití v automobilu FS02.

Seznam odborné literatury:

[1] Freescale Semiconductor, Inc., MPC5200 Hardware Specification, 2005
[2] NXP B.V., LPC2109/2119/2129 Single-chip 16/32-bit microcontrollers, 2007
[3] Robert Bosch GmbH, CAN Specification, 1991

Vedoucí diplomové práce: **Ing. Pavel Burget, Ph.D., K13135**

Datum zadání diplomové práce: **červen 2009**

Platnost zadání do[1]: **leden 2011**

Prof. Ing. Miloš Mazánek, CSc.
vedoucí katedry

Doc. Ing. Boris Šimák, CSc.
děkan

Praha, leden 2010

[1] Platnost zadání je omezena na dobu tří následujících semestrů.

Obsah

1 Úvod....1
2 Diagnostika pro závodní automobily....2
3 Senzory....5
3.1 Senzor detekce natočení volantu....5
3.1.1 Optické senzory....5
3.1.2 Resolver....6
3.1.3 Dvou fázový MRE....7
3.2 Senzor polohy tlumičů....7
3.3 Teplotní senzor....9
3.4 Senzor otáček kol....9
3.4.1 Induktivní senzor....9
3.4.2 Senzor s Hallovým jevem....10
3.4.3 Optický senzor....11
4 Výběr senzorů....12
5 Procesorový modul PPC5200....16
5.1 HW....16
5.1.1 Procesor MPC5200CVR400B....17
5.1.2 Operační paměť....18
5.1.3 Bootovací paměť....18
5.1.4 Paměť FRAM....18
5.1.5 Systém reálného času (RTC)....19
5.1.6 Ethernet....19
5.1.7 Sériová rozhraní....19
5.1.8 Napájení modulu....20
5.1.9 Konektory....20
5.2 SW....20
6 Rozhodnutí o použití senzorického modulu s procesorem LPC21xx....22
6.1 Procesor LPC21xx....22
6.2 Rozhodnutí....22
7 Návrh hardwaru pro procesorový modul a procesor LPC2119....24
7.1 Obvody....24
7.1.1 Zdroj napětí....24
7.1.2 Resetovací obvody....25
7.1.3 CAN převodník....26
7.1.4 USB převodník....27
7.1.5 Sériová linka....27
7.1.6 Záloha času....27
7.1.7 A/D převodník....27
7.1.8 Spínací tranzistor....28
7.1.9 Detekční diody....28
7.2 Návrh plošného spoje....28
7.3 Programování procesorů....30
7.3.1 Úprava programátoru....31
7.3.2 Zapojení programovacího konektoru....31
8 Implementace softwaru pro využití v automobilu FS02....33
8.1 Nahrání programu LPC21xx....33

8.2 GPIO....33
8.3 Linux....35
8.3.1 Nastavení....35
8.4 Vytvoření programu....37
8.4.1 Program pro ukládání dat....37
8.4.2 Program pro spojení s počítačem....39
9 Realizace....41
10 Zhodnocení....46
Reference:....47
Přílohy i
A Zapojení konektoru procesorové desky MPC5200 i
B Schéma desky MIDAM_FS02....ix
C Tabulka součástek použitých ve schématu pro desku MIDAM_FS02....x
D Schéma zapojení programátoru....xiv
E Obsah DVD....xv

Seznam obrázků

Obrázek 2.1: STACK 800[1]....2
Obrázek 2.2: Data logger Bosch C Sport[2]....3
Obrázek 2.3: ACL [3]....4
Obrázek 3.1: Optický detektor [4]....5
Obrázek 3.2: Resolver [5]....6
Obrázek 3.3: Dvoufázový MRE senzor [4]....7
Obrázek 3.4: Tlumič s implementovaným senzorem [6]....8
Obrázek 3.5: Potenciometr - senzor polohy tlumiče [7]....8
Obrázek 3.6: Optický senzor pro měření vzdálenosti [8]....9
Obrázek 3.7: Induktivní senzor - princip [9]....10
Obrázek 3.8: Hallův jev [10]....10
Obrázek 3.9: Optické senzory[11]....11
Obrázek 4.1: Senzor polohy tlumiče....12
Obrázek 4.2: Senzor natočení volantu [12]....13
Obrázek 4.3: Senzor otáček kol v FS02....14
Obrázek 4.4: GPS EM-411 - Engine Modul [14]....14
Obrázek 4.5: GPS NL-303P - PDA [14]....15
Obrázek 5.1: Blokové schéma modulu shark [15]....16
Obrázek 5.2: Blokový diagram procesoru MPC5200 [17]....18
Obrázek 5.3: Multiplexování periférií [17]....19
Obrázek 5.4: Procesorová deska s procesorem MPC5200 od firmy Microclima s.r.o. [18] 20
Obrázek 7.1: Zapojení sběrnice CAN s procesorem [21]....26
Obrázek 7.2: Senzor a schématické znázornění zapojení....28
Obrázek 7.3: Náhled na vrchní stranu tištěného spoje....29
Obrázek 7.4: Náhled na dolní stranu tištěného spoje....30
Obrázek 7.5: Programovací konektor pro LPC2119....31
Obrázek 8.1: Nabíhání procesorového modulu 34
Obrázek 8.2: Poslání zprávy a přijmutí zpráv po sběrnici CAN....36
Obrázek 8.3: Program pro síťové spojení s počítačem (odpojený stav)....39
Obrázek 8.4: Program pro síťové spojení s počítačem (spojený stav)....40
Obrázek 9.1: Svrchní strana desky....42
Obrázek 9.2: Spodní strana desky....42
Obrázek 9.3: Testování a dolaďování záznamové desky....43
Obrázek 9.4: Ochranný box prázdná....44
Obrázek 9.5: Ochranný box s modulem....44
Obrázek 9.6: Ochranný box ze shora....45
Obrázek 9.7: Ochranný box zespodu....45

1 Úvod

V roce 1981 spolek automobilových inženýrů (SAE) rozběhl ve Spojených státech amerických program Formula SAE. jedná se o konstrukční soutěž, která má za cíl vyrobit malé závodní auto určené pro omezenou sériovou výrobu. V Evropě, konkrétně ve Velké Británii se soutěž konala v roce 1998. Tým CTU CarTech z Českého vysokého učení technického v Praze, který se doposud zabývá stavbou závodního vozu, byl založen v roce 2007 a já jsem se stal jeho členem o rok později.

Pro vyvíjený vůz bylo potřeba navrhnout záznamové zařízení, které by zaznamenávalo průběh chování vozu při závodu. Získaná data by pak byla použita pro optimalizaci konceptu nového vozu. Návrhem a realizací diagnostického systému pro CTU CarTech FS02 se zabývá tato práce.

2 Diagnostika pro závodní automobily

Na trhu je k nalezení mnoho výrobců zařízení záznamu dat z automobilu. Několik výrobců a jejich výrobky zde uvedu. Vybíral jsem je podle dostupnosti na trhu, kladného ohlasu uživatelů a použitelnosti pro vůz FS02.

STACK 800 je od firmy ISA-RACING [1] základní informace uvedu v pár bodech:

- Jedinečná flexibilita, díky modulární konstrukci
- Snadno přizpůsobitelné novým úkolům prostřednictvím rozšiřujících modulů
- Nejvyšší bezpečnost systému podle normy MIL-Norm- a MSS-Stecker, teplotně odolné kabely a hardwar, který byl testován s vibracemi 20 G
- Bezpečné ukládání dat do kvalitní paměti
- Bezkonkurenční rychlost přenosu dat prostřednictvím přenosové soustavy STACK
- Snadné ovládání ze strany fronty a softwaru DataPro DESIGNER
- Unikátní zobrazení a analýza obrazu
- Snadná konfigurace pomocí drag and drop a pull-down menu
- Nejlepší funkci zobrazení trasy na trhu. Komplikované post-processing není nutná
- Všechny systémy včetně software - žádný čas-omezený licencí

Obrázek 2.1: STACK 800[1]

K danému přístroji lze připojit až 25 drátových vstupů, má integrovaný převodník sběrnice CAN, možnost přenosu dat přes W-LAN síť, rozšířitelné o GPS modul a možnost zabezpečení pomocí PIN. Modulově lze rozšiřovat o nejrůznější snímače senzorů. Například modul MF20, který disponuje s 12 analogových vstupů (0-5V), 8 impulsních, má implementovaný senzor zrychlení (3 osy).

Data Logger C Sport je od firmy BOSCH. Tento výrobce je velice znám z automobilového průmyslu, kde dodává elektroniku do běžných aut. Data Logger C Sport (obrázek 2.2) je velice lehký (210g), umí zpracovat 4 analogové diferenční, 4 analogové jednoduše zakončené a 4 digitální vstupy [2]. Disponuje s 512 MB interní pamětí, data od ECU přijímá přímo po CAN sběrnici, dále je vybaven další CAN sběrnici pro příjem dat z vozu, síťový konektor a hodiny reálného času.

- CAN (sběrnice pro spojení s ECU)
- CAN sběrnice (pro propojení s jednotkami C Sport)
- síťový konektor (pro komunikaci s počítačem)
- 512 MB vnítřní paměti
- hodiny reálného času
- 4 digitální vstupy
- 4 analogové vstupy s 16 bit rozlišením
- 4 analogové diferenciální vstupy s 16 bit rozlišením
- 1 digitální výstup

Obrázek 2.2: Data logger Bosch C Sport[2]

ACL od firmy MoTeC (obrázek 2.3) je nastavitelný pro sběr dat a komunikaci. ACL provádí načítání dat, datové komunikace a sofistikované výpočty. Získává data z jiných zařízení, jako jsou MoTeC ECU, Dash záznamníku a až osm VIMS, které mu umožňují záznam více než 200 vstupů. To poskytuje všechny pokročilé funkce ADL3, včetně varování, alarmy, předpověď paliva, motor logy, časovače, uživatelské podmínky a telemetrie. Systém se údajně jednoduše nastavuje a všechny jejich jednotky se řídí z jedné softwarové aplikace.

- 1 GB datové paměti
- stahování přes síťové rozhraní
- rychlé zaznamenávání dat až 5000 vzorků za sekundu
- Kombinované zaznamenávání vyšší než 20 MB / min
- Data do ACL jsou vnesena pomoci periférií, ty mohou zahrnovat VIMS, E888, E816, SDL, PLM, ECU, a GPS
- 2 x CAN s individuálně programovatelným rychlosti sběrnice CAN
- 1 x RS232, 1 x RS485

Obrázek 2.3: ACL [3]

3 Senzory

Senzorů pro diagnostiku je celá řada. Proto se zaměřím jen na ty, které se dají použít v automobilu. Potřebné měřené veličiny jsou: poloha vozu, poloha natočení volantu, poloha tlumičů, teplota motoru, palubní napětí, otáčky motoru, rychlost otáčení kol.

3.1 Senzor detekce natočení volantu

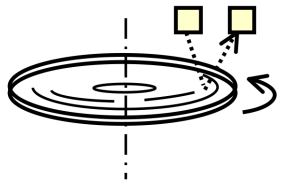
Senzorů natočení volantu je mnoho, všechny jsou však na podobném principu. Většina má vlastní vyhodnocovací jednotku, která vyhodnotí v jakém úhlu natočení se daný senzor nachází. Tyto data zformuje a pošle přes sběrnici CAN.

3.1.1 Optické senzory

(čtení barkódu)

- citlivé na vysoké teploty
- citlivé na špínu
- drahé

Hlavní součástí senzoru je kotouč, který funguje podobně jako CD (obrázek 3.1). Na ploše jsou místa, která po osvícení světlo odrazí a místa která světlo neodrazí. Takto je na disku zakódován barkód jejž je specifický svou vlastností, že každé sousední kódové slovo je změněno pouze o jeden bit, tím se zvyšuje spolehlivost.



Obrázek 3.1: Optický detektor [4]

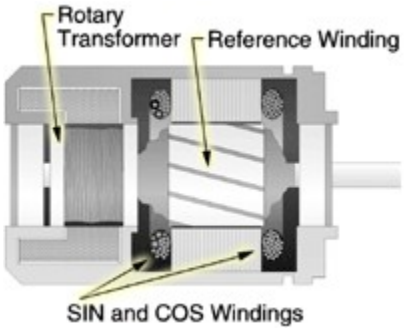
3.1.2 Resolver

(indukování napětí, dvou fázová detekce)

- veliké rozměry
- drahé

Resolver je analogové čidlo, které bylo původně určeno pro vojenské účely. Je to otočný transformátor měnící sinusové napětí v závislosti na otáčející se hřídeli. Skládá se z primárního, referenčního vinutí a ze dvou sekundárních vinutí – sinusové a kosinusové (obrázek 3.2). Reference vinutí se nachází v rotoru resolveru, sinusové a kosinusové vinutí ve statoru. Sekundární vynutí jsou navzájem od sebe posunuté o 90°. Energie do referenčního vynutí je dodávána přes rotační transformátor, tím se eliminují kartáčky a sběrací kroužky.

Indukované napětí v sekundárních vinutí jsou rovná hodnotě referenčního napětí vynásobeného sin nebo cos úhlu hřídele z nulového bodu. Takto získáme dvě napětí jejichž podíl představuje absolutní pozici hřídele.



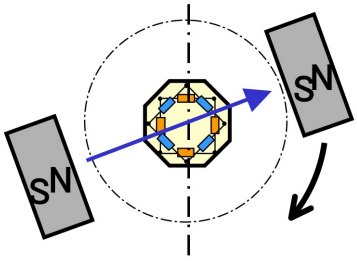
Obrázek 3.2: Resolver [5]

3.1.3 Dvou fázový MRE

(otáčení v magnetickém poli, dvoufázová detekce)

- u levnějších je nemožné detekovat 180° až 360°

Tento senzor je dosti podobný předchozímu principu. Měří se vzájemná poloha elektromagnetického pole. Má spirálově zatočenou hlavní cívku v osmiúhelníku, dva mosty sinusových cívek a dva mosty kosinusových cívek. Tím se zajistí možnost detekovat 0° do 360°.



Obrázek 3.3: Dvoufázový MRE senzor [4]

3.2 Senzor polohy tlumičů

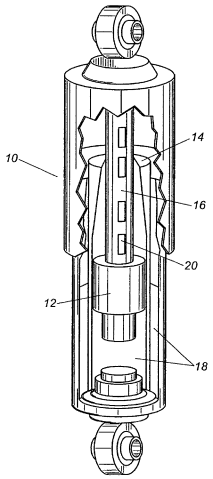
Měřit odpružení vozu lze pomocí senzorů implementovaných přímo v tlumiči, externě připojených, nebo pomocí měření vzdálenosti vozovky od nápravy.

První varianta znamená výrobu tlumiče již se senzorem. U vnitřních senzorů se jedná o použití permanentních magnetů a malé cívky – poloha je snímána skokově, není tedy možno rozlišit, v jaké části tlumiče se nacházíme, jen jeho posun. Nebo lze použít stupnici s různou permitivitou (obrázek 3.4), potom lze poznat, v jaké poloze se tlumič nachází. Výroba takového tlumiče je jiná a ne všichni výrobci jej dokážou vyrobit. Při výrobě na zakázku je pořizovací hodnota o mnoho vyšší než by byla za samotný tlumič + senzor zvlášť.

Dále je možnost umístit senzor paralelně s tlumičem. Jedná se o potenciometr, který mění svůj odpor v závislosti na velikosti jeho vysunutí/zasunutí (obrázek 3.5). Tímto senzorem lze zjistit, v jaké vzdálenosti se tlumič nachází. Některé typy mají integrovaný A/D(1) převodník, u něj jsou získávána data o velikosti vysunutí senzoru. Jinak je potřeba mít A/D převodník pro získání dat do digitální formy.

(1) A/D – převodník z analogového signálu na digitální

Optický senzor (obrázek 3.6) je bezkontaktní. Umisťuje se na konstrukci vozidla nebo na karoserii s laserovým paprskem směřujícím kolmo na zem. Snímač je schopen přesně měřit a monitorovat světlou výšku vozidla s četností až 500 krát za sekundu (500Hz), také je schopen pracovat s mnoha povrchy včetně lesklé gumy a obsahuje jednotku, která zpracovává a řídí měření. Zpracovaná data jsou v digitální podobě. V závislosti na zvolené referenci, vůči které se měří (pneumatika, země) je započítáno pružení celého vozu, případně i pneumatik.



Obrázek 3.4: Tlumič s implementovaný m senzorem [6]

Obrázek 3.5: Potenciometr - senzor polohy tlumiče [7]

Obrázek 3.6: Optický senzor pro měření vzdálenosti [8]

3.3 Teplotní senzor

Senzor pro sledování teploty je téměř v každém sériově vyráběném motorovém vozidle. Jeho nepostradatelnost tkví v tom, že díky němu lze předejít zničení motoru přehřátím. Senzor sleduje teplotu chladící kapaliny vozidla. V případě nebezpečně veliké teploty varuje obsluhu a předá informaci do řídící jednotky, což bývají to termistory (teplotně závislé odpory) nebo teplotně závislé diody (speciální křemíkové diody s lineárním teplotní závislostí na odporu).

3.4 Senzor otáček kol

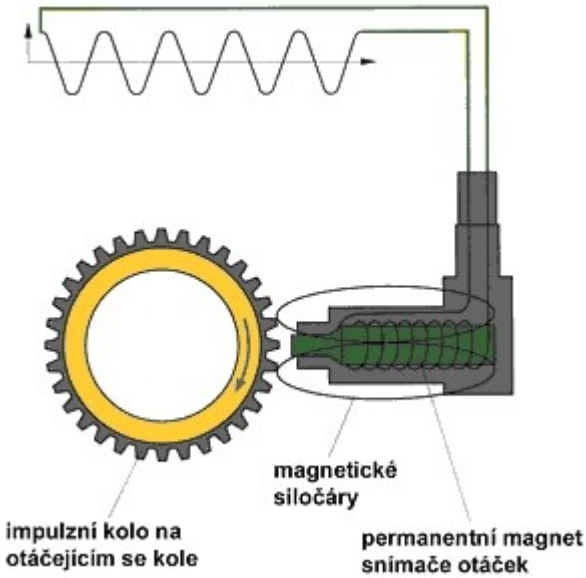
Tento senzor bude na zjištění aktuální rychlosti vozu, protáčení kol při rozjezdu, při průjezdu zatáčkou a zpětné analýzy chování podvozku na vozovce.

Pro zjišťování rychlosti otáčení kol je dobré mít více jak jeden senzor na celém vozu, zpřesní se tím měření otáček a veličin z toho vyplývajících (rychlost, ujetá dráha). Nepřesnosti by mohly vznikat například při vjezdu vozu do zatáčky, kdy je jiná rychlost vnitřních a vnějších kol, nebo při projíždění zatáčkou přetáčivě, či nedotáčivě. Také lze zjistit momenty, kdy se protáčejí kola s náhonem.

Senzorů je v praxi používáno několik:

3.4.1 Induktivní senzor

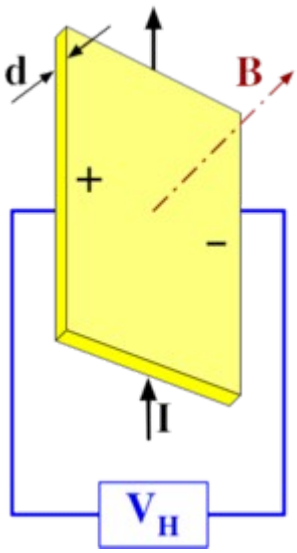
Detektor tvoří cívka s jádrem z permanentního magnetu, do které se indukuje napětí. Obrázek 3.7 zobrazuje princip snímače. Na otáčejícím se kole je umístěno feromagnetické kolo s přesně vyřezanými dílky. Tyto dílky způsobí, že při otáčejícím se kole se bude měnit napětí na senzoru, podle náčrtku os v horní části obrázku. Z těchto změn napětí se dá poznat, jak rychle se kolo otáčí, neboť je znám počet zubů (změn napětí) za jedno otočení kola a je znám jeho obvod. Tento senzor má však velikou nevýhodu – špatně rozeznává pomalé otáčení kol. Pomalé otáčení způsobí malou změnu pole za čas. Tím se nenaindukuje dostatečné veliké napětí a není možné jej odlišit od šumů.



Obrázek 3.7: Induktivní senzor - princip [9]

3.4.2 Senzor s Hallovým jevem

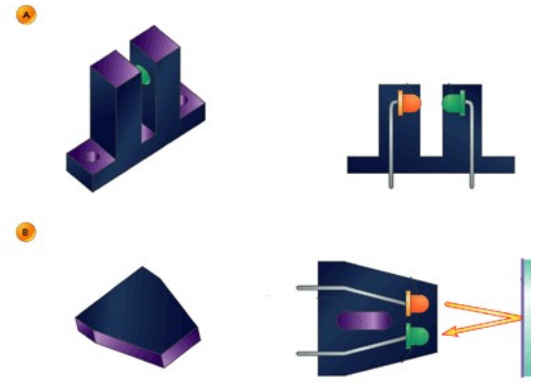
Hallův jev objevil Edwin Hall v roce 1879 [10]. Díky tomuto jevu jsme schopni měřit intenzitu magnetického pole, jehož princip je vidět na obrázku 3.8. Pomocí konstantního proudu I se objevuje na svorkách napětí (Hallovo) závislé na intenzitě magnetického pole, do kterého je detektor vložen. Jako senzor je používán s dírkovaným feromagnetickým diskem. V senzoru je vedle detekční destičky umístěn permanentní magnet. Tento magnet přes feromagnetický disk zvyšuje magnetické pole v okolí detektoru, případně snižuje, pokud se nachází na místě, kde je disk. Děje se to v závislosti zda před detektorem je feromagnetická látka nebo není.



Obrázek 3.8: Hallův jev [10]

3.4.3 Optický senzor

Senzor tvořený svítivou diodou a detekční diodou je další možný způsob detekce otáček. Používá se například v kuličkových myších. Pro detekci pohybu se používá také dírkovaných kotoučů (pro senzor na obrázku 3.9A) u kterých ovšem nezáleží na materiálu, z něhož jsou vyrobeny, nebo lze použít kotouč s odraznou plochou (pro senzor na obrázku 3.9B) v níž budou bezodrazová místa. Senzor má jednu velikou nevýhodu, je potřeba zajistit čistou plochu zdroji světla a detektoru.



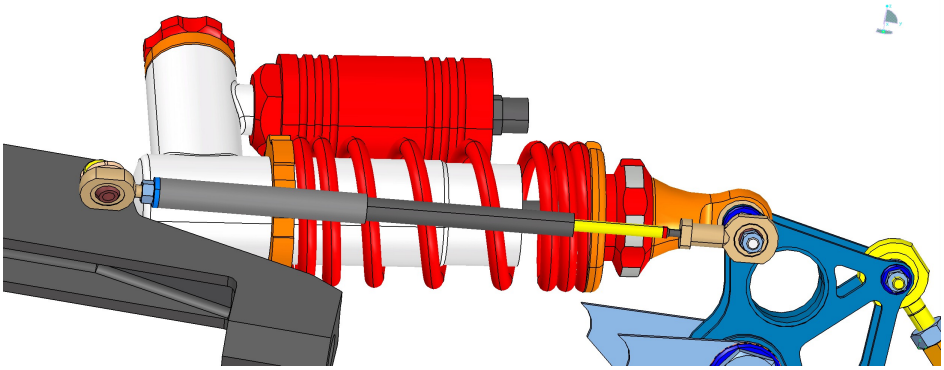
Obrázek 3.9: Optické senzory[11]

4 Výběr senzorů

V předešlé kapitole o dostupných záznamových zařízení jsem psal, že je možnost připojit různé senzory. Těch je nepřeberné množství a je vždy důležité vědět, jaké veličiny jsou potřeba měřit. V případě vozu FS02 se tým dohodl, že bude potřeba sledovat teplota motoru, poloha klapky, palubní napětí, sledování tlumičů, natočení kol, rychlost a v poslední řadě i poloha vozu.

V řídící jednotce je spousta údajů o vozidle, například palubní napětí, teplota motoru, clutch master a poloha klapky. Tyto informace se dají v případě závodní ECU zjistit na sběrnici CAN.

Pro sledování polohy tlumičů jsou nejvhodnější senzory připojitelné vně tlumičů. Tato varianta je výhodná z hlediska mnoha faktorů, ale tlumiče jsou již vybrány a implementování senzorů do hotového výrobku by bylo dost nákladné a dost technicky náročné. Zbyly tedy dvě varianty jak měřit polohu odpružení. Měření pomocí optického snímače je přesné a rychlé, oproti souběžně připojenému odporovému senzoru má ovšem nevýhody. Při provozu v prašném prostředí je nebezpečí zanesení senzoru, který potřebuje dobrou optickou viditelnost. Pro svoji funkci potřebuje přímou viditelnost na vozovku a neměří přímo pohyb tlumiče. Z těchto důvodů se nejvíce hodí odporový pohybový senzor. Jednoduše se přidá ke každému tlumiči (obrázek 4.1) a po kabelu se bude přenášet analogový signál přímo do jednotky, kde se bude zpracovávat. Je potřeba převést v jednotce analogový signál na digitální.



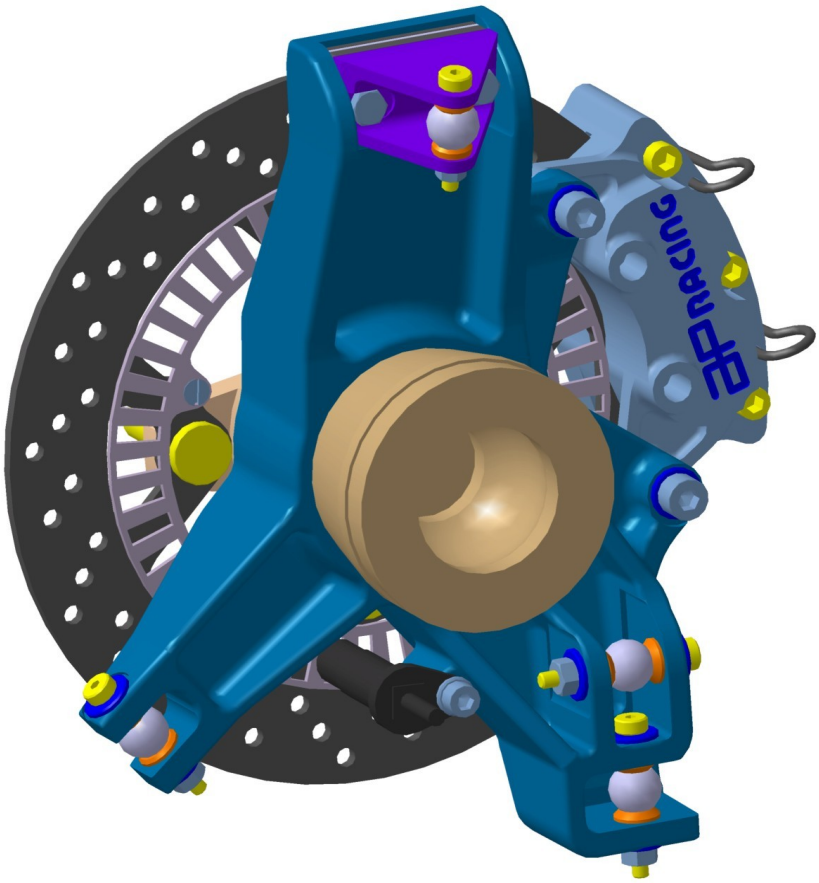
Obrázek 4.1: Senzor polohy tlumiče

Senzor natočení volantu (obrázek 4.2) je velice rozšířený detektor. V dnešní době ho má téměř každý automobil. Starší senzory mívali digitální TTL výstup. Dnes mají přímo výstup na sběrnici CAN. V případě implementace to velice ulehčuje situaci. Jednotka, která zpracovává údaje je umístěna přímo u senzoru, a dále jsou distribuována pouze data v digitální podobě. Tento senzor vyrábí firma Bosch.

Obrázek 4.2: Senzor natočení volantu [12]

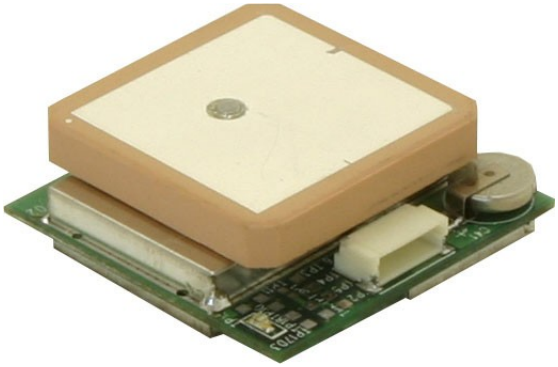
Pro vůz FS02 je senzor teploty převzat ze sériově vyráběného motocyklu Yamaha R6 (r.v. 2008). Tento senzor je sveden do řídící jednotky, kde je vyhodnocena hodnota teploty. Teplota bude brána z řídící jednotky, která ji posílá po sběrnici CAN.

Pro snímání otáčení kol byl zvolen senzor s hallovým jevem [13]. Na každém kole je umístěn disk s vyfrézovanými dírami a senzor (obrázek 4.3). Signál ze senzoru je veden do jednotky kontroly trakce, kde je signál vyhodnocen a přepočítán na rychlost. Tato rychlost je posílána po sběrnici CAN. Tato jednotka také poskytuje informaci při každé celé otáčce jednotlivých kol, pro zaznamenávání ujeté vzdálenosti.



Obrázek 4.3: Senzor otáček kol v FS02

Mnoho záznamových zařízení ukládá pozici, to je umožněno velikým rozšířením GPS modulů. Ty jsou dostupné a přesné. Potřebný GPS modul by měl být co možná nejméně poruchový a dostatečně citlivý. Rozhodoval jsem se mezi mnoha zařízeními. Jako



Obrázek 4.4: GPS EM-411 - Engine Modul
[14]

vhodný GPS modul se ukázali EM-411 - Engine Modul (obrázek 4.4) a NL-303P - PDA Empfänger (obrázek 4.5). Komunikují přes sériovou linku (rs 232) a základním hardwarem

se nijak neliší, oba mají čipset SiRF Star III. Druhý zmíněný má výhodu, že je dodáván současně se softwarem a je umístěn v prachotěsném obalu. Ale je o čtvrtinu ceny dražší než zmíněný první modul.

Obrázek 4.5: GPS NL-303P - PDA [14]

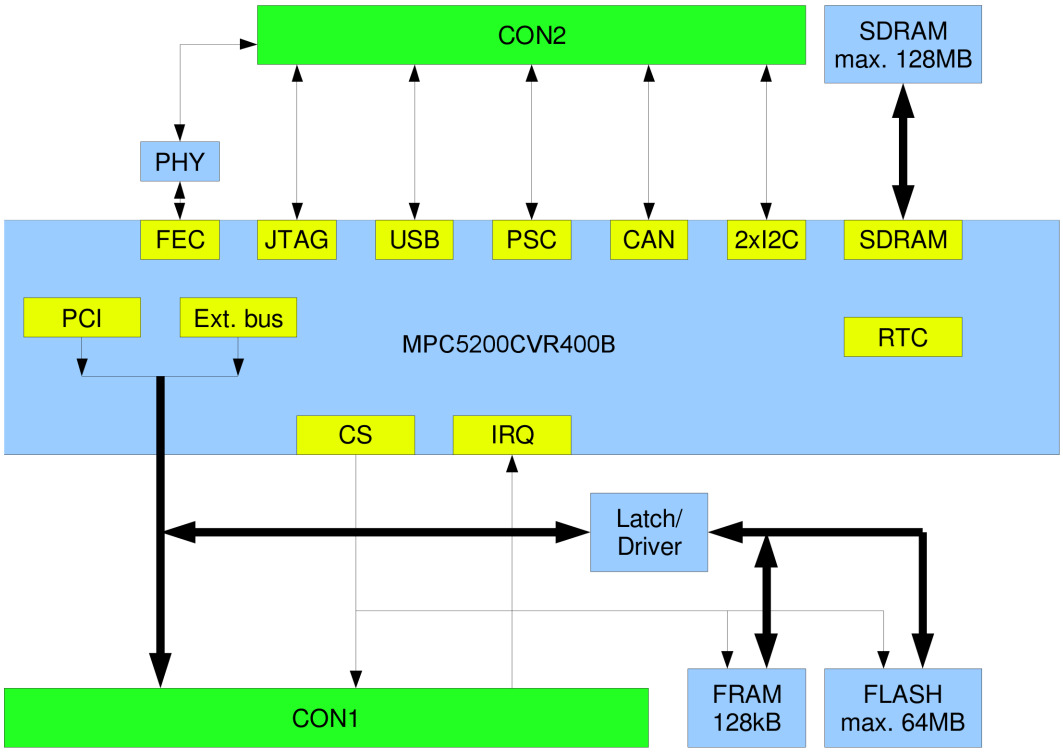
5 Procesorový modul PPC5200

5.1 HW

Používaný procesorový modul má v sobě výkonný procesor PowerPC s jádrem 603e od společnosti Freescale semiconductor. Jeho typové označení je: MPC5200 a označení součástky: MPC5200CVR400B. Procesor má takt jádra 396 MHz (12 x 33 MHz) a výpočetní výkon 760MIPS, takt procesoru zajišťuje programovatelný oscilátor 33 MHz.

Procesor se vyrábí jen v pouzdru TE-PBGA 272 pinů, 27 x 27 mm. Koupili jsme jej již jako součást modulu Shark V1.1 [15]. Kde jsou na desce plošných spojů umístěny obvody pro správný chod samotného procesoru (obrázek 5.1) :

- procesor MPC5200B
- operační paměť – 2x SDRAM Samsung (32M x 16)
- Bootovací paměť
- Paměť FRAM
- Systém reálného času (RTC)



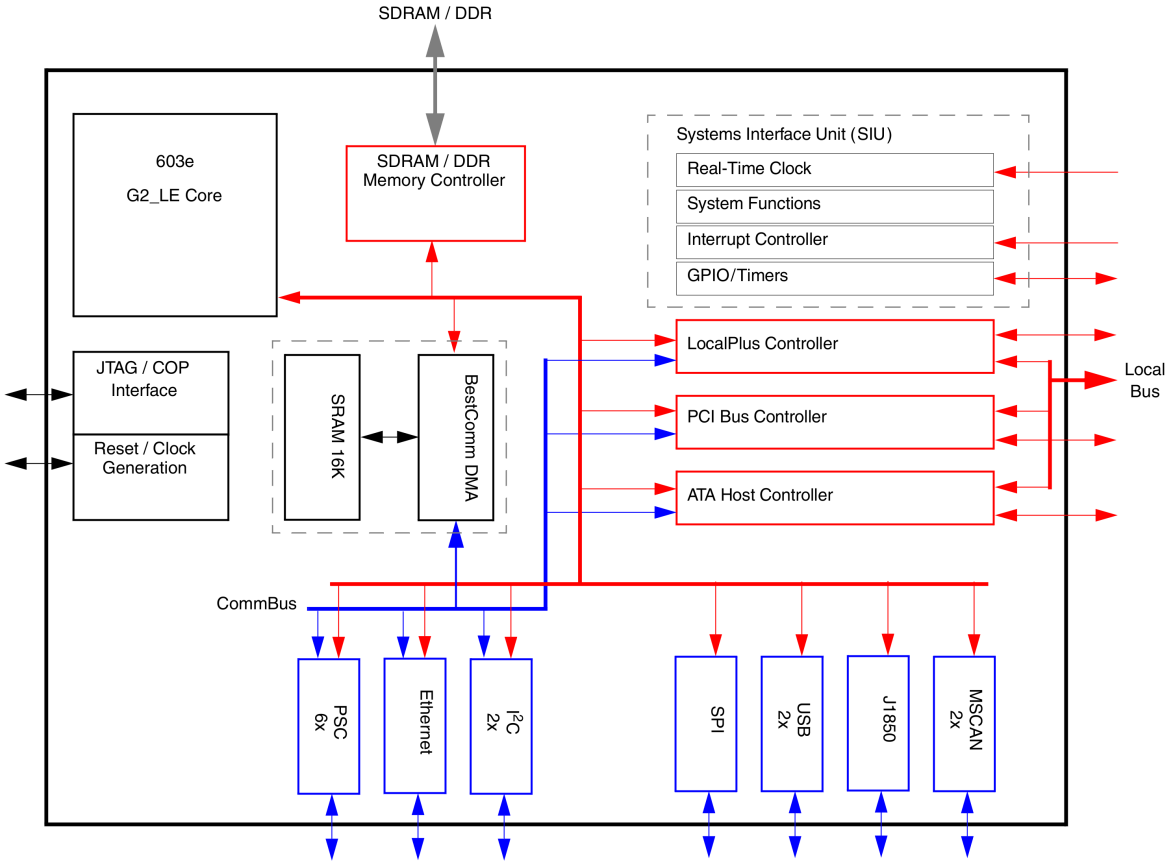
Obrázek 5.1: Blokové schéma modulu shark [15]

5.1.1 Procesor MPC5200CVR400B

Procesor disponuje mnoho vlastnostmi, nejdůležitější zde uvedu [16]:

- MPC603e série Power Architecture™ procesor
- 0-400 MHz, při -40°C až +85°C
- Double Precision Floating Point Unit (FPU)
- Instruction and Data Memory Management Unit (MMU)
- 16K Instrukční a 16K data caches
- BestComm Intelligent DMA I/O Controller
- SDR a 133 MHz DDR paměti (266 MHz)
- Local Plus interface for flash memory
- 10/100 Ethernet MAC
- Peripheral Control Interface (PCI) Version 2.2
- ATA/IDE Interface
- USB 1.1 Host (2x, USB 2.0 compatible)
- Programmable Serial Controllers (6x)
- Serial Peripheral Interface (SPI)
- I^2C (2x)
- I^2S (až 3x)
- CAN 2.0 A/B (2x)
- J1850 BDLC-D
- GPIO (do 56)
- 8 časovačů
- 1.5V core, 3.3V external (and 2.5V for DDR memory)
- 272 Pin Plastic Pin Ball Grid Array (PBGA) Package

Jeho blokový diagram je zobrazen na obrázku 5.2. Zde je vidět mnoho periférií, tyto periférie bohužel nelze úplně všechny využívat najednou, neboť kapacita vývodů to nedovoluje. Proto je v procesoru registr GPIO, kterým se nastavuje, které periférie použiji a na jakých vývodech budou.



Obrázek 5.2: Blokový diagram procesoru MPC5200 [17]

5.1.2 Operační paměť

Modul je osazen dvěma paměťmi SDRAM s kapacitou 64MB zapojenými paralelně. Přístup do operační paměti je realizován 32-bitovou datovou sběrnicí a 13-bitovou adresovou sběrnicí (multiplexovaná řádková a sloupcová adresa).

5.1.3 Bootovací paměť

Procesor bootuje z paměti FLASH o velikosti 64MB. Připojení paměti FLASH je realizováno pomocí externí multiplexované sběrnice v konfiguraci 16 - bitů pro data, 26 - bitů pro adresu. Paměť FLASH je připojená na CS0. Tuto paměť je možné případně odpojit odebráním jednoho nulového odporu na desce modulu a bootovat tak z externí paměti.

5.1.4 Paměť FRAM

Modul je osazen jednou paralelní pamětí typu FRAM od firmy Fujitsu MB85R1001 o velikosti 128kB připojenou na externí sběrnici. Paměť FRAM je přístupná pomocí CS1.

5.1.5 Systém reálného času (RTC)

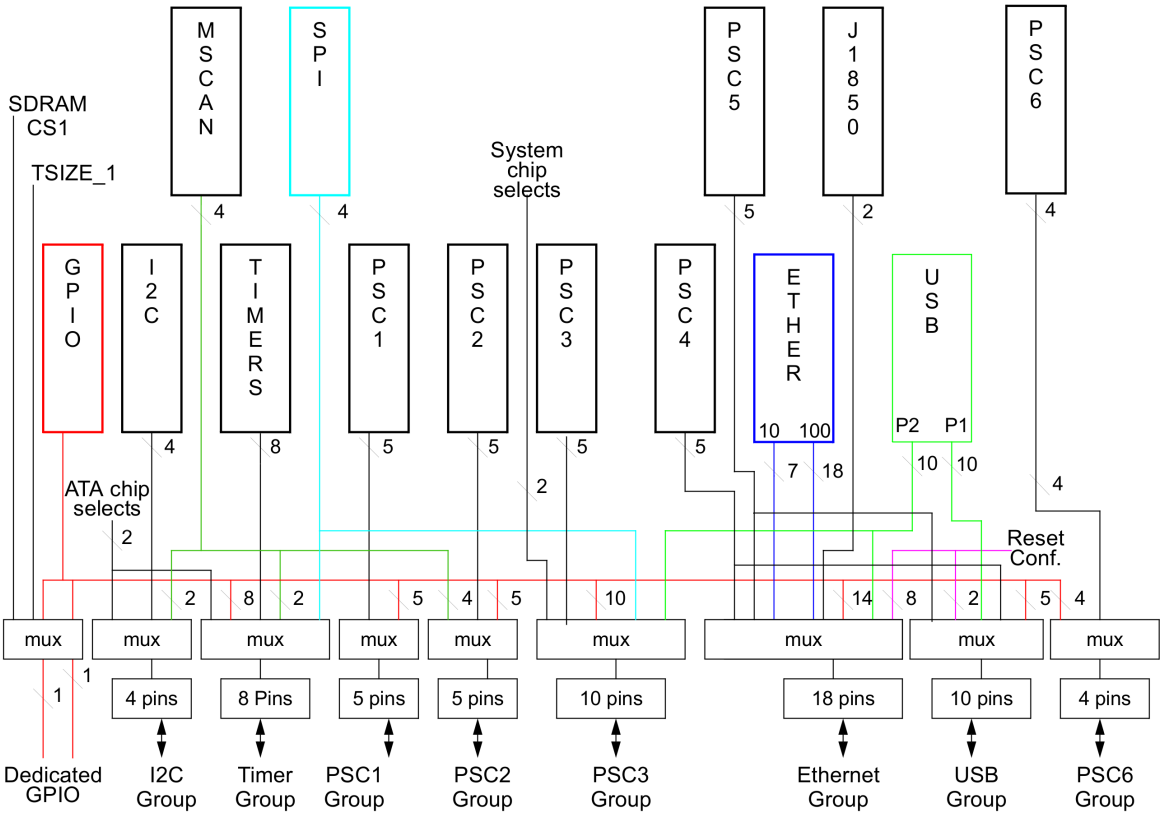
Na desce modulu je osazen krystal 32,768kHz pro taktování reálného času systému. Obvod RTC je integrován přímo v procesoru MPC5200. Pro chod hodin je nutné trvalé připojení napájení +3,3V. V případě nutnosti zálohy reálného času musí být použit externí RTC obvod se záložní baterií připojený k modulu např. přes sériové rozhraní I2C.

5.1.6 Ethernet

Na desce modulu je fyzická vrstva Ethernetu SMSC LAN8700I-AEZG. Připojení fyzické vrstvy k procesoru je po standardní sběrnici MII. Signály pro připojení externího síťového transformátoru jsou vyvedeny na konektoru CON2. Schéma zapojení séťového transformátoru lze najít v dokumentech na webových stránkách výrobce (LAN8700(I) MII Customer Evaluation Board, +3.3V I/O VDDIO Operation, Schematic Revision E0P1).

5.1.7 Sériová rozhraní

Procesor MPC5200 má softwarově programovatelné sériové rozhraní (PSC), které lze nastavit na různé druhy sériové komunikace - UART, AC97, USB2, SPI, IrDA, CAN,



Obrázek 5.3: Multiplexování periférií [17]

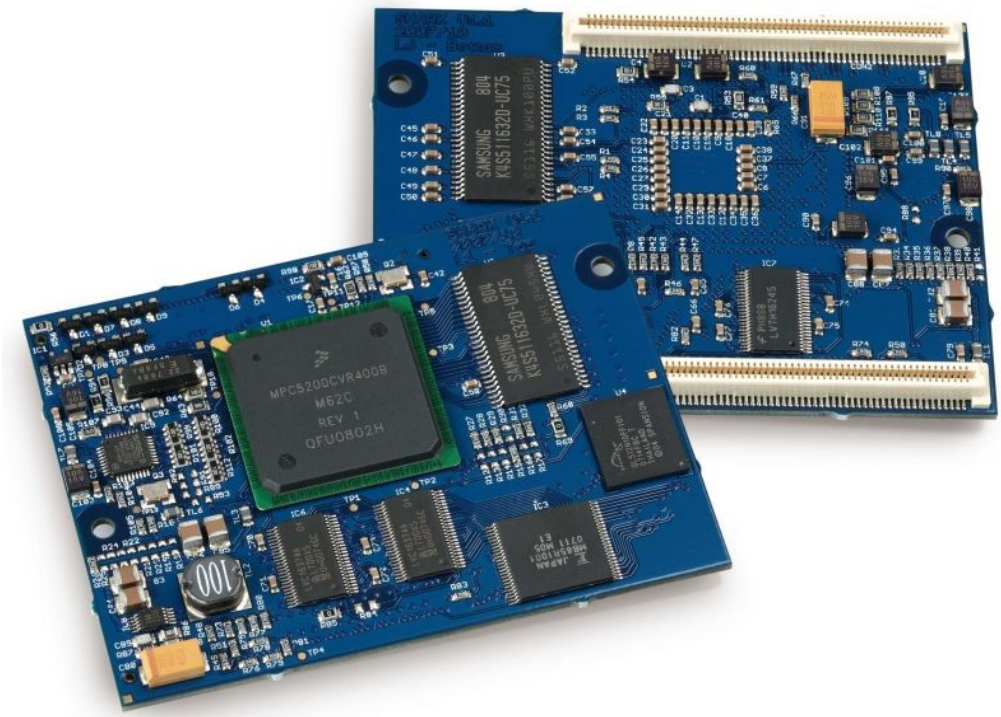
nebo jako GPIO. Všechny brány PSC jsou vyvedeny na konektor. Možnosti módování jednotlivých bran PSC jsou vysvětleny na obrázku a v [17].

5.1.8 Napájení modulu

Napájecí napětí modulu je +3,3V s tolerancí ±5%. Napájení jádra procesoru je řešeno
spínaným stabilizátorem +3,3V na +1,5V.

5.1.9 Konektory

Modul je osazen dvěma konektory Board to Board firmy AMP, typové označení 177983-5 (viz obr. 5.4). Každý z nich má 120 pinů. Dohromady má deska 240 pinů. Tabulka zapojení těchto pinů pro navrhovanou desku je v příloze A.



Obrázek 5.4: Procesorová deska s procesorem MPC5200 od firmy Microclima s.r.o. [18]

5.2 SW

Na procesor byl implementován U-Boot pro inicializaci procesoru a jeho periferií, který zároveň slouží jako zavaděč operačního systému nebo uživatelského softwaru.

Podporovanými operačními systémy jsou Linux s verzí jádra 2.6 nebo PikeOS firmy Sysgo.

Použitý modul má nahraný operační systém Linux 2.6.26.5. Ke kterému se dá připojit pomocí síťového rozhraní nebo propojením se sériovým portem modulu. Mnou použitý OS lze stáhnout ze zdroje [15].

6 Rozhodnutí o použití senzorického modulu s procesorem LPC21xx

Před samotným navrhováním schémat a zapojení obvodů jsem se musel rozhodnout jak bude vypadat celý senzorický modul. Byly na výběr dvě možnosti, buď využít desku se samotný procesorovým modulem nebo senzorického modulu s procesorem LPC21xx. Celý modul musí zvládnout zpracovat signály od všech senzorů.

Potřebné periférie pro zpracování signálů a zápis dat:

- sběrnice CAN
- USB
- rs 232
- 4 analogové vstupy

Z popisu procesorového modulu vím, že obsahuje sběrnici CAN, USB, rs 232 avšak neobsahuje analogové vstupy. Bylo by tedy potřeba externího A/D převodníku.

6.1 Procesor LPC21xx

Procesor je s jádrem ARM7 vyráběn firmou NXP Semiconductors [19]. Má rozhraní I^2C, SPI, UART CAN, 4 vstupy 10 bitového A/D převodníku a další vlastnosti, pro mě nezajímavé.

Tento procesor obsáhne ve vlastní flash paměti až program s daty do velikosti 128kB. Procesor je po naběhnutí schopen samostatné činnosti.

6.2 Rozhodnutí

Jako nejlepší varianta se zdála implementace A/D převodníku přímo na podpůrnou desku pro modul MIDAM. Pro jeho připojení je nutné rozhraní I^2C, to procesor podporuje, ale není implementováno v softwaru. Jeho implementování by znamenalo napsání vlastní knihovny pro práci s I^2C v Linuxu a předělání jádra Linuxu. Napsání knihovny a její implementace do systému by trvala nezkušenému programátorovi velmi dlouho. Proto tuhle variantu nepoužiji, ale pokusím se ji nějak obměnit. Pokud by data chodila přímo na sériový port, tak by se jádro systému nemuselo měnit.

Druhá varianta je již zmíněný senzorický modul. Jako základ pro tento modul by byl použit procesor LPC21xx s kterým mají dobré zkušenosti na katedře řídící techniky.

Modul má implementovaný 4 A/D převodníky, nemusel bych přidávat již žádné další obvod. Dále má mnoho výhod, jeho asi největší výhoda je jeho otestování a odzkoušení v provozu. Již v loňském roce byl použit v automobilu FS01.

Ze zde uvedených možností se mi jevila druhá možnost mnohem lepší a časově méně náročná. Zvolená varianta má výhodu dvou nezávisle běžících systémů.

7 Návrh hardwaru pro procesorový modul a procesor LPC2119

Jako první jsem si rozepsal nároky, které byli kladeny na hardware, aby byl návrh efektivní.

Pro modul desky PowerPC je potřeba zajistit:

- zdroj napětí 3,3 V stejnosměrný
- resetovací obvod závislý na velikosti napájecího napětí
- CAN převodník
- USB převodník (USB)
- RS232 převodník (sériovou linku)
- obvod pro zálohování času
- síťový konektor

Pro procesor LPC2119 je potřeba zajistit:

- zdroj napětí
    - 3,3 V stejnosměrný
    - 1,8 V stejnosměrný
- resetovací obvod závislý na velikosti napájecího napětí
- CAN převodník
- krystal
- obvody pro A/D vstup procesoru
- spínací tranzistor

7.1 Obvody

Všechny obvody byly zapojeny podle doporučeného schéma zapojení v dokumentaci dané součástky. Zapojení jsou na schématu v příloze B.

7.1.1 Zdroj napětí

Hardware musí být použitelný do vozu formule, kde je umístěna 12 V baterie o celkové kapacitě 8,6 Ah. Baterie je dobíjena přes alternátor od motoru. Třífázové napětí z alternátoru je přivedeno do regulátoru, kde se usměrní a vyrobí napětí. To se používá pro

napájení elektrických zařízení vozu a dobíjení baterie. Napětí za regulátorem v provozu je kolem 14 V až 14,5 V. Minimální hodnota bude přibližně 11,4 V, to je napětí úplně baterie.

Výrobce procesorového modulu uvádí odebíraný proud cca 500 mA, mnou naměřený odběr na testovací desce RYU_edu(1) byl kolem 250 mA, při spuštěném běhu všech potřebných periférií. Procesor LPC2119 odebírá do 10 mA. Tuto hodnotu jsem změřil na modulu LpcEurobot(2). Ze součtu vychází hodnota 260 mA pro chod procesorů. Při připočítání proudů od integrovaných obvodů vnějších periférií, indikačních diod a usb flash disku vychází výsledný proud s malou rezervou do 500 mA. Za podmínky, že souběžně půjdou všechny obvody.

Palubní napětí se tedy pohybuje přibližně od 11 do 15 V a odebíraný proud nebude vyšší než 500 mA. V tomto rozmezí musí být deska schopna korektně fungovat. V obvodech jsou potřeba napájecí napětí 5 V, 3,3 V a 1,8 V. Jako nejlepší variantu jsem vyhodnotil použití spínaného napěťového regulátoru, který sníží vstupní napětí na 5 V. Využil jsem integrovaný obvod LM2575, u něj je zapotřebí čtyř externích součástek[20] (cívka, dioda, 2x kondenzátor), výstupní proud až 1A. Pro další stabilizaci jsem vybral LDO(3) stabilizátor, který bude z 5V měnit snižovat napětí na 3,3 V a 1,8 V. Na 3,3 V je to obvod LM1117 (až 800 mA), na 1,8V obvod TC1017 (až 150 mA).

7.1.2 Resetovací obvody

Pro případ, že by se napájecí napětí snížilo pod minimální napájecí hodnotu, je nutné procesor externě resetovat a držet ho v resetu, dokud napětí nenabude správných hodnot. Pokud by tam nebyl takovýto obvod, hrozilo by nebezpečí nekorektního chování procesoru (na některých vstupech by se mohla změnit logická hodnota 1 na logickou hodnotu 0). Nebezpečí hrozí také při zapnutí napájení, kdy se hodnota napájecího napětí postupně zvyšuje.

Pro oba procesory nesmí jmenovité napětí 3,3 V klesnout pod 3,0 V, zvolil jsem tedy obvody, které při poklesu napájecího napětí pod minimální napájecí napětí resetují procesor.

(1) RYU_edu – deska plošných spojů vyrobená pro výukové účely na katedře řízení.
(2) LpcEurobot – deska s procesorem LPC2119 s periferiemi a napájecím obvodem.
(3) LDO – low dropout voltage (nízkoúbytkový stabilizátor).

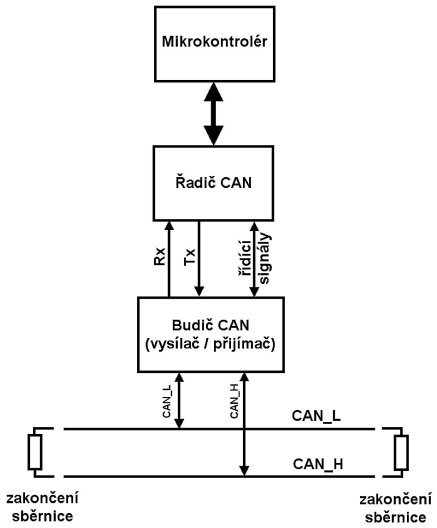
Zvolil jsem pro každý procesor jiný resetovací obvod:

- **MAX811** od firmy MAXIM je obvod s ručním resetem. Tento vstup se hodí při ladění programu. Je připojen k procesorovému modulu a přes pull-up rezistor na 3,3 V. Spojení vstupu reset s pull-up resitorem mi zaručí, že zde nebudou nedefinované stavy. Při uvedení resetu do logické 0 bude na vstupu logická 0, jinak tam bude logická 1 díky pull-up rezistoru.
- **MCP130** od firmy MICROCHIP je zapojen ke vstupu reset na procesoru LPC2119 a zapojen přes pull-up rezistor na 3,3 V.

7.1.3 CAN převodník

Pro připojení sběrnice CAN k procesorovému modulu a procesoru LPC2119 je potřeba budiče CAN (obrázek 7.1). Použil jsem obvod SN65HVD230.

Pro správnou funkci sběrnice je potřeba mít vždy po dvou odporech, na každé straně sběrnice jeden. V případě připojení více periférii na sběrnici mohou zůstat pouze dva odpory, více by zkomplikovalo přenos. Záznamové zařízení není jako jediné na sběrnici CAN, proto je na ní umístěn odpor, ale tak aby mohl být v případě potřeby jednoduše odpojen. K tomu slouží přepínač SW-DIP04.



Obrázek 7.1: Zapojení sběrnice CAN s procesorem [21]

7.1.4 USB převodník

Pro funkčnost USB na procesorovém modulu je potřeba USB budič a obvod řídící napájení portu USB napětím 5 V. Pro tuto aplikaci jsem vybral obvody USB1T11A (budič USB) a MIC2025 (řízení napájení portu USB).

7.1.5 Sériová linka

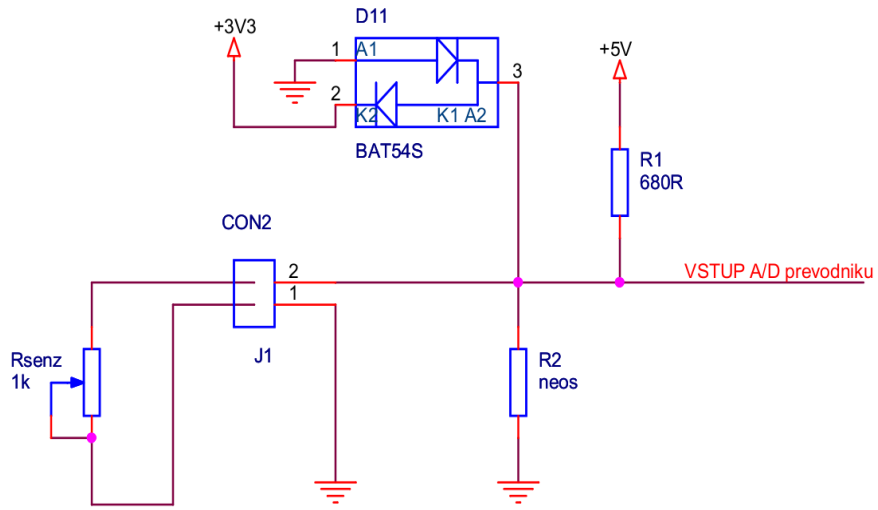
Pro možnost připojení sériové linky je potřeba mít převodník, který umí pracovat s logickými úrovněmi 3,3 V, proto jsem vybral MAX3232. Zapojil jsem ho se 4 externími kondenzátory, které slouží jako nábojové úložiště při komunikaci.

7.1.6 Záloha času

Při vypnutí napájení si procesorový modul neuchová reálný čas. V této aplikaci je reálný čas důležitou informací při zpětném prohlížení dat. Abychom jej mohli mít a používat, je nutné si čas zálohovat. Pro zálohování jsem vybral obvod M41T00 se zálohovaným napájením z baterie CR2032, takt určuje připojený krystal s frekvencí 32,768 kHz. Komunikuje s procesorem přes sběrnici I^2C. Čas v procesoru je načtený vždy při zapnutí procesorového modulu. Při zapnutém procesorovém modulu lze také nastavit čas v zálohovacím obvodu.

7.1.7 A/D převodník

Senzory odpružení a senzor otáčení volantu (při uvažování jiného senzoru než jsem uvedl ve výběru) jsou pouze odporového charakteru. Pro jejich implementaci je potřeba údaj rezistivity převést na napětí. To lze udělat pomocí odporového děliče, kterým teče proud v závislosti na proměnném odporu (senzoru). Odpor R1 omezuje proud při Rsenz = 0 na 7 mA. Tento stav by měl nastat výjimečně. Odpor R1 je spočten na 680 Ω, aby mohlo být využito maximálního rozmezí A/D převodníku v procesoru. Jako ochrana jsou zde umístěny dvě schotkyho diody, pro případ, že by na vstup bylo připojeno zápornější napětí nebo vyšší napětí než je 3,6 V. Pro nedostatek volných vstupů A/D převodníku u procesoru jsem použil analogový multiplexor ADG804.



Obrázek 7.2: Senzor a schématické znázornění zapojení

7.1.8 Spínací tranzistor

Pro možnost spínat větší proudy (accusump) jsem umístil na desku spínací tranzistor MOSFET(60 V, 30 A, 70 W). Je připojen na vývod z procesoru LPC2119 a je připojen na společnou zem s kostrou vozu. Přívodní drát je připojen do jedno-pinového konektoru. Tranzistor spíná na kostru vozu, stačí tedy přivést pouze jeden kabel, dále jsou zde zapojeny ochranné diody proti přepětí.

7.1.9 Detekční diody

Pro snazší odlaďování, oživování a kontrolu jsem umístil na modul 5 detekčních svíticích diod. Dle schématu v příloze B označené diody je D2, D7, D8, D9 a D16.

Jednotlivé diody značí:

- D2 indikuje běh zdroje napájení
- D16 indikuje napájení USB portu
- D7 až D9 jsou připojeny na výstup z procesoru LPC2119 a po naprogramování lze s nima indikovat běh programu.

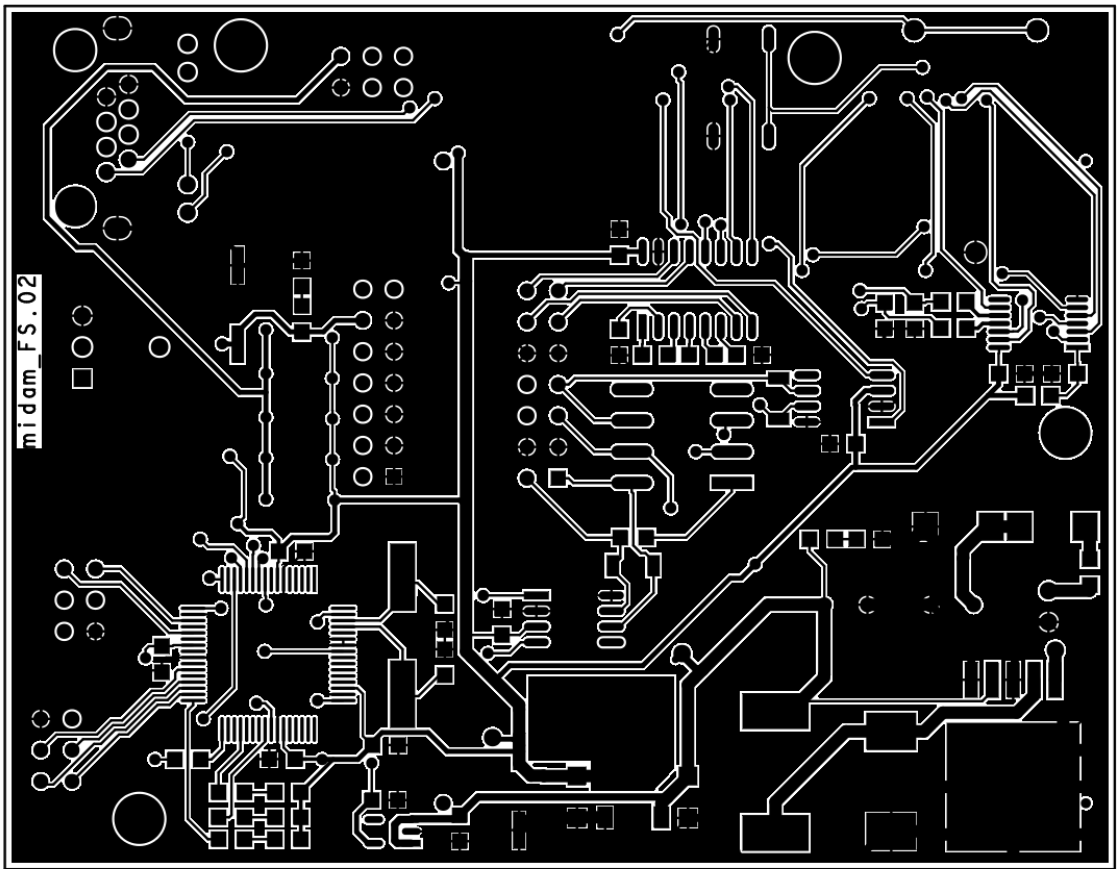
7.2 Návrh plošného spoje

Největší nároky pro návrh plošného spoje byly jeho rozměry. Deska měla být co možná nejmenších rozměrů. Navrhl jsem jí o 1 cm větší než je velikost procesorového

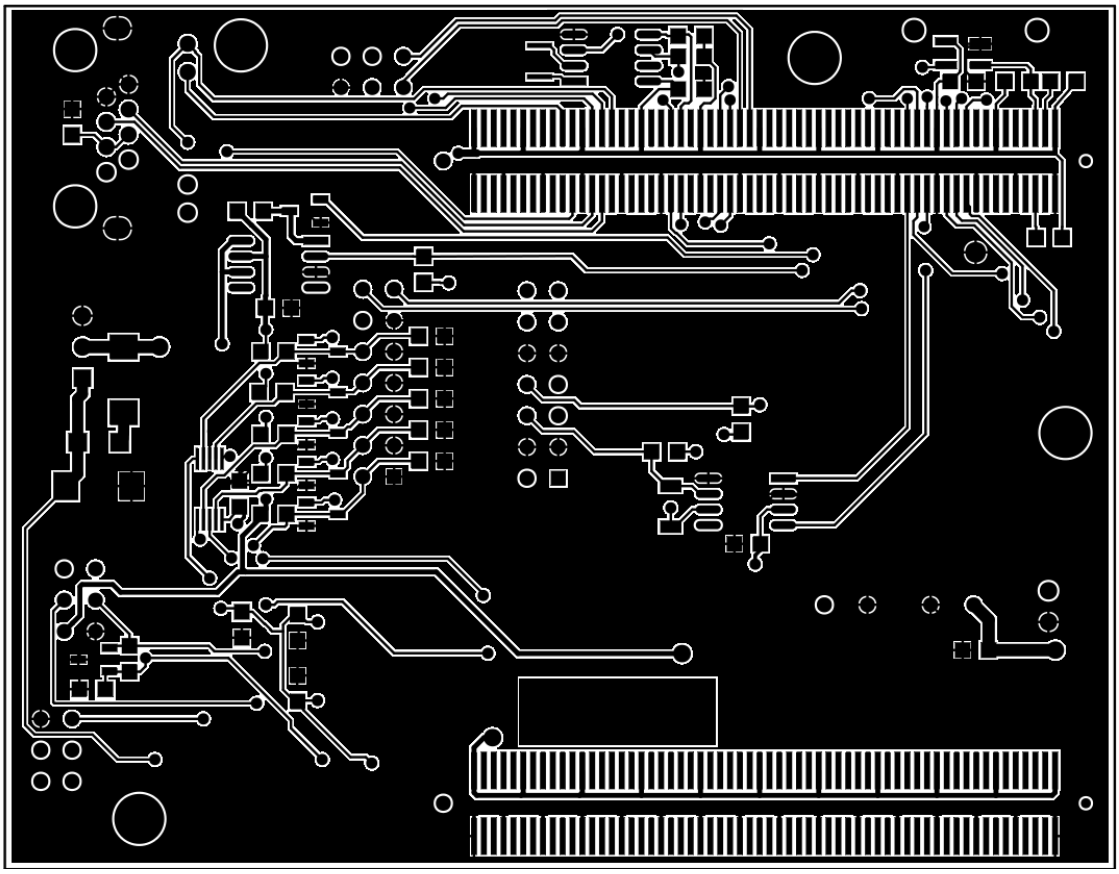
modulu, tedy v rozměru 90 x 70 mm.

Plošný spoj pro oba procesory jsem navrhoval v programu OrCAD. Zvolil jsem dvoustranou desku s osazováním součástek po obou stranách. Konektory pro připojení vnějších periférii a většinu objemnějších součástek jsem umístil na horní stranu desky (TOP). Na spodní stranu desky jsem umístil konektory pro procesorový modul a pár menších součástek. Takovéto rozložení jsem zvolil, abych si uvolnil místo na plošném spoji, kde bude připojen procesorový modul. Tomu nesměly vadit žádné součástky v zasunutí konektoru a také byla potřeba minimální mezery pro průchod vzduchu. Dále jsem vytvořil malou chladící plošku z obou stran desky spojenou prokoveními pro obvod LM1117.

Podklady pro výrobu plošného spoje jsou umístěny na přiloženém DVD a náhled na plošný spoj je na obrázcích 7.3 a 7.4.



Obrázek 7.3: Náhled na vrchní stranu tištěného spoje



Obrázek 7.4: Náhled na dolní stranu tištěného spoje

7.3 Programování procesorů

Pro nahrání programu do procesorů je potřeba sériového rozhraní na počítači. Bohužel v posledních letech se běžné počítače nevybavují sériovou linkou. Aby bylo možné procesory naprogramovat z jakéhokoliv počítače s běžným rozhraním, jsem se rozhodl pro převodník z USB na sériovou linku. Rozhodoval jsem se mezi dvěma možnostmi umístění převodníku. Buď byla možnost dát jej přímo na navrhovanou desku, nebo mimo na nějakou jinou destičku. První varianta má výhodu jedné komponenty, není potřeba ničeho dalšího pro práci s deskou. Druhá má více výhod: použije se pouze jednoho převodníku a tím se ušetří náklady na potřebné obvody (cca 1/12), ušetří místo na desce plošných spojů, tak je možné mít menší desku. Využil jsem již hotový převodník USB – programátor LPC [22] a trochu jej předělal dle vlastních potřeb (schéma viz. Příloha D).

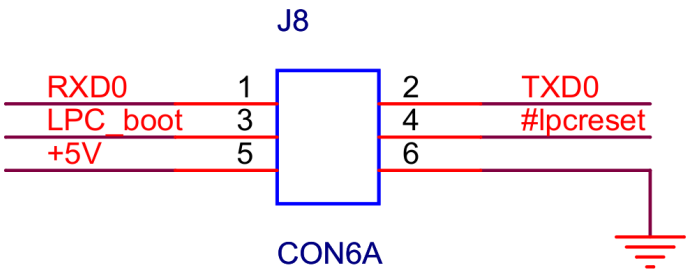
7.3.1 Úprava programátoru

Na schématu programátoru v práci [22] (příloha B4) je vidět původní zapojení. Pro nahrávání programu jsem musel přerušit spoj FTD R# (šestý pin obvodu U1 a pátý pin konektoru J2) a připojit na pátý pin konektoru J2 druhý pin obvodu U1 (pin reset na procesoru). Pro možnost napájení z programátoru jsem přerušil spoj před devátým pinem konektoru J2 a propojil jej přes schotkyho diodu. Diodu jsem zde umístil pro případ, že je zapnuté napájení programované desky. V takovém případě se bude napájet ze zdroje pro desku s procesory, napájení z USB bude mít nižší hodnotu o přechodové napětí diody (~ 0,3 V) a dioda zůstane zavřená. Pro napájení obvodu U1 jsem propojil na propojce J3 piny 2 a 3. Část se spínaným zdrojem a tlačítky jsem nevyužil, pro aplikaci nebyly potřeba.

7.3.2 Zapojení programovacího konektoru

Pro správné zapojení programátoru je důležité znát zapojení konektoru na desce plošného spoje. Ten je zobrazen na obrázku 7.5. Jsou zde piny pro naprogramování procesoru LPC21xx:

- RXD0 a TXD0 jsou linky sériového rozhraní procesoru, po kterém jdou data (piny 1 a 2 konektoru J2)
- LPC_boot je vstup pinu kterým dáme procesoru najevo, že do něj chceme nahrát data (pin 3 konektoru J2)
- #lpcreset je vstup vstupu pro reset procesoru, který je po dobu programování aktivní (pin 5 konektoru J2
- značka ukostření je společná svorka GND pro celou desku (pin 10 konektrou J2)
- +5V je pin pro napájení z programátoru 5 V (pin 9 konektoru J2)



Obrázek 7.5: Programovací konektor pro LPC2119

Pro sériovou komunikaci s procesorovým modulem je konektor stejný, jen u něj

nejsou potřeba piny LPC_boot a #lpcreset.

Procesorový modul lze programovat přes síťové rozhraní pomocí ssh. Je to jednodušší, rychlejší než přes sériové rozhraní a také nabízí více možností. Lépe se sleduje běh programu, testuje nový program a snáz se přijde na případné chyby.

8 Implementace softwaru pro využití v automobilu FS02

Aby celá soustava fungovala, je potřeba, aby procesory měly svůj program. Tento program je potřeba napřed napsat a následně nahrát. Tomuto je věnována tato kapitola.

Pro psaní programu používám běžný počítač s operačním systémem Linux 2.6.35 distribuci Ubuntu 10.04 a využívám programu pro psaní textu Kate. Textový editor má vlastnosti rozlišení programovacích jazyků a zvýraznění příkazů. Program jsem napsal v jazyce C a ke kompilaci jsem využil Ocera Make Systém (OMK), který je používán přes konzoli (příkazový řádek). Přesný popis zacházení a stažení je dostupný v [23] .

Pro kompilaci programů je nutné mít nainstalované balíky:

pro LPC21xx:

- binutils-arm-elf
- gcc-arm-elf

pro procesorový modul

- binutils-powerpc-linux-gnu_2.18.0-1
- gcc-powerpc-603e-linux-gnu_4.1.1
- gcc-powerpc-linux-gnu_4.3.2-1
- glibc-powerpc-linux-gnu_2.7-1

případně novější balíky. S těmito mám vyzkoušeno, že to funguje.

V poslední řadě je důležité mít nainstalován sériový terminál, použil jsem gtkterm.

8.1 Nahrání programu LPC21xx

K nahrání programu jsem využil OMK. V konzoli jsem se dostal k umístění programu, napsal „make“ a program se zkompiloval. Připojil jsem programovací desku k počítači a k desce s procesorem na konektor J8, příkazem „make load-flash“ se nahrál program do flash paměti procesoru.

8.2 GPIO

Před spuštěním desky si je potřeba rozmyslet, jak jsou rozmístěny spoje na konektoru k procesorovému modulu. Aby fungovalo vše jak má, musí mít registr GPIO řečeno, jaká rozhraní je potřeba mít na konektoru. Přesnou hodnotu jsem se dočetl v

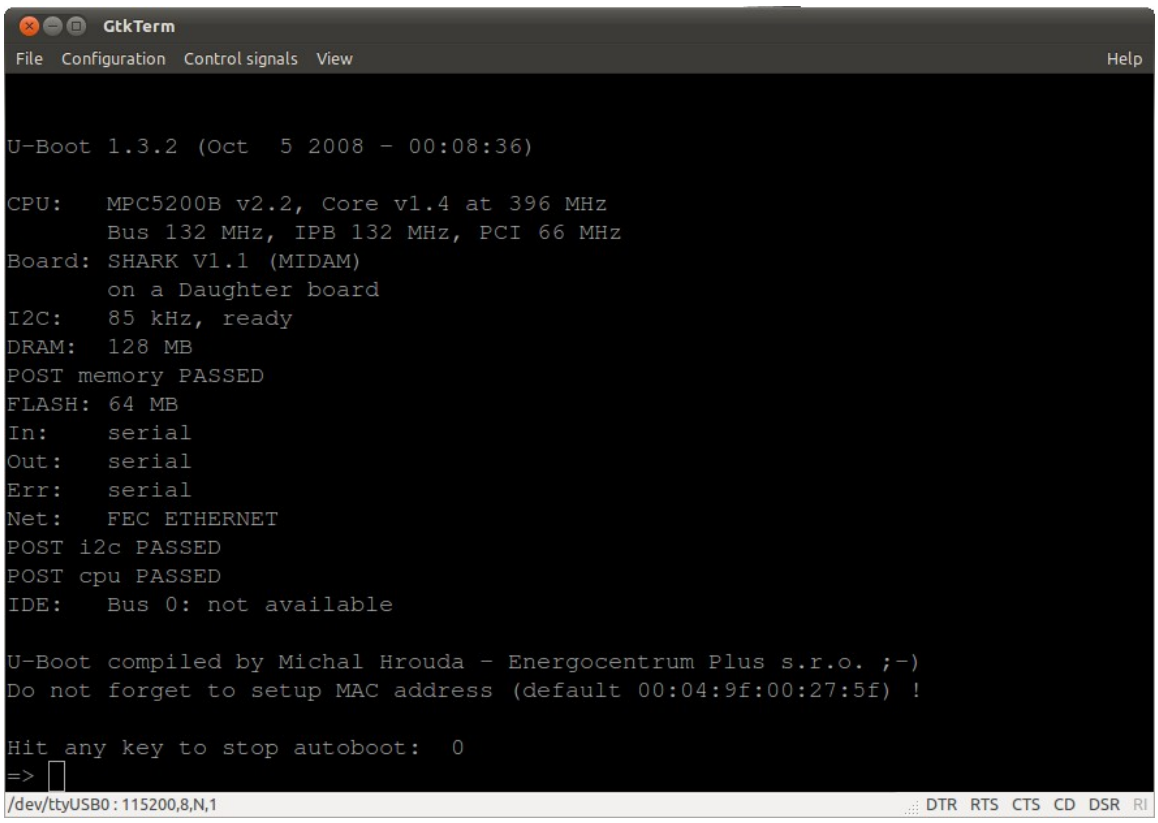
dokumentaci k procesoru [16]. Hodnota je: 0x91551044.

Tento údaj je nutné zadat v Uboot před načtením systému. Provádí se to pouze, pokud je konektor nastaven jinak. Tedy pouze jednou při používání jedné desky.

Do Uboot jsem se dostanu pomocí programu gtkterm. Zapul jsem napájení desky, připojil k desce programátor a zapojil jej do počítače. Gtkterm nastavil následovně:

- baud rate: 115200 bps
- bits: 8 bit
- stop bits: 1 bit
- Parity: none
- Flow control: none
- port: připojené USB - /dev/ttyUSB0

Po restartu procesoru jsem se dostal do Uboot, stisknutím klávesy jsem přerušil načítání OS (obrázek 8.1). Gpio jsem přečetl příkazem „printenv psc_cfg“, pokud by byla hodnota jiná, tak bych jí nastavil příkazem „set psc_cfg 91551044“ a uložil „saveenv“. V Uboot je uložena potřebná hodnota pro desku a mohl jsem nechat načíst OS.



Obrázek 8.1: Nabíhání procesorového modulu

8.3 Linux

8.3.1 Nastavení

Při prvním zprovoznění jsem nevěděl, jakou bude mít modul IP adresu, tak jsem se na něj nemohl připojit přes síťové rozhraní. Nechal jsem načíst OS systém a byl jsem připojen přes sériové rozhraní. Nastavil jsem IP a připojil jsem síťový kabel. Na počítači jsem nastavil IP adresu a pomocí ssh protokolu se připojil s přihlašovacím jménem „root“ na desku. Abych nemusel nastavovat IP po každém zapnutí, přidal jsem řádku do souborového systému na desce /etc/init.d/rc.local (lokální nastavení) „ifconfig eth0 10.0.3.2 netmask 255.255.255.0“ nastaví mi IP adresu modulu na 10.0.3.2.

Při psaní programu jsem měl připojenou desku k počítači a pomocí NFS jsem sdílel složku se zkompilovanými zdrojovými kódy. Bylo to rychlejší než jej po každé kopírovat na desku. Aby se vše načetlo automaticky při naběhnutí systému, tak jsem připsal dalších pár řádek do /etc/init.d/rc.local

„mount -t nfs -o nolock 10.0.3.4:/home/mpc/build/ppc/_compiled/bin /home/mnt“

připojí z IP adresy 10.0.3.4 složku zkompilovaných zdrojových kódů do /home/mnt na procesorrovém modulu.

Ještě bylo potřeba připojit flash disk na který se budou ukládat data. Přidaný řádek je: „mount /dev/sda1 /mnt/lexar/“ - připojí disk sda1 (flash disk pres USB) do /mnt/lexar/.

Pro zprovoznění čtení ze sběrnice CAN bylo nutné ji inicializovat. To jsem provedl příkazem „caninit“. Operační systém má jako výchozí hodnotu pro přenos po CAN sběrnici nastavenou rychlost 1000000. Moduly připojené na CAN0 májí tuto rychlost, ale na CAN1 mají tuto rychlost poloviční, tedy 500000, která je daná centrální řídící jednotkou auta. Nastavení rychlosti jsem docílil vložením hodnoty 500000 do souboru /sys/class/net/can1/can_bittiming/bitrate. To jsem provedl příkazem „echo 500000 > /sys/class/net/can1/can_bittiming/bitrate“. Do souboru lokálního nastavení jsem přidal 4 příkazy:

```
caninit
ifconfig can1 down
echo 500000 > /sys/class/net/can1/can_bittiming/bitrate
ifconfig can1 up
```

Při změně rychlosti sběrnice bylo nutné ji napřed deaktivovat, proto příkaz „ifconfig can1 down. Následně byla zapnula příkazem „ifconfig can1 up“.

Dále bylo třeba spustit samotný program, který byl umístěn vždy na připojovaném flash disku. Spustil jsem jej příkazem „ ./mnt/lexar/program &“. Ten jsem také zařadil do souboru lokálního nastavení.

OS má příkazy, kterými lze číst a zapisovat data z rozhraní . U CAN to je příkaz candump pro přijetí a cansend pro odeslání zprávy. Mají vlastní formát, který zjistíme přidáním „ -h“ za příkazy. Je nutné specifikovat, na které sběrnici se má příkaz provést například „candump can0“, viz obrázek 8.2.

```
root@shark ~> cansend can0 5a1#12.2542
root@shark ~> candump can0
  can0    3  [6] 62 62 63 64 65 00
  can0    4  [6] 67 67 68 69 6A 00
  can0    5  [6] 6C 6C 6D 6E 6F 00
  can0    6  [6] 71 71 72 73 74 00
  can0    7  [6] 76 76 77 78 79 00
  can0    8  [6] 7B 7B 7C 7D 7E 00
  can0    9  [6] 80 80 81 82 83 00
  can0    A  [6] 85 85 86 87 88 00
  can0    B  [6] 8A 8A 8B 8C 8D 00
  can0    C  [6] 8F 8F 90 91 92 00
  can0    D  [6] 94 94 95 96 97 00
  can0    E  [6] 99 99 9A 9B 9C 00
  can0    F  [6] 9E 9E 9F A0 A1 00
  can0   10  [6] A3 A3 A4 A5 A6 00
  can0   11  [6] A8 A8 A9 AA AB 00
  can0   12  [6] AD AD AE AF B0 00
  can0   13  [6] B2 B2 B3 B4 B5 00
  can0   14  [6] B7 B7 B8 B9 BA 00
  can0    1  [6] BC BC BD BE BF 00
  can0    2  [6] C1 C1 C2 C3 C4 00
  can0    3  [6] C6 C6 C7 C8 C9 00
  can0    4  [6] CB CB CC CD CE 00
  can0    5  [6] D0 D0 D1 D2 D3 00
  can0    6  [6] D5 D5 D6 D7 D8 00
^Croot@shark ~>
```

Obrázek 8.2: Poslání zprávy a přijmutí zpráv po sběrnici CAN

Pokud je spuštěna celá deska poprvé, je nutné nastavit hardwarové hodiny reálného času, které běží na desce plošných spojů i po odpojení napájení. Provádí se to napsáním příkazu „hwclock“. Nejprve je potřeba vložit správné datum a čas do systému linux, to se provede pomocí příkazu „date YYYY.MM.DD-hh:mm:ss“, kde *YYYY* je rok, *MM je* měsíc, *DD* je den, *hh* jsou hodiny, *mm* jsou minuty, *ss* jsou sekundy. Je nutné dodržet formát a počet číslic, je možné zadat čas i jinak, více je v nápovědě příkazu „date -h“. Po správném zadání se vypíše v konzoli čas a datum, lze si jej ověřit napsáním „date“. Když máme nastaveno, převede se čas do hardwarových hodin pomocí „hwclock -w“. Nastavenou hodnotu si lze překontrolovat příkazem „hwclock -r“, více o příkazu po zadání „hwclock -h“.

8.4 Vytvoření programu

Program by měl umět získat z periférií potřebné informace, upravit je do srozumitelné formy, uložit na záznamové zařízení připojené k desce a případně bezdrátově předávat do počítače.

8.4.1 Program pro ukládání dat

Pro získávání informací z periférií, pro mě již jen ze sběrnice CAN, jsem využil socket CAN. Programuje se podobně jako síťové rozhraní. Je nutné ho inicializovat na začátku programu a poté jen pomocí funkce *select* čtu. Po přečtení informace se pomocí *case* rozřadí informace do určitého souboru, zapíše a případně bezdrátově odešle.

Celý program je umístěn v jedné nekonečné smyčce, ta obsahuje inicializaci a spuštění procesu čtení z periférií. Inicializace je nutná pro správné načtení periférií a pro možnost používat je. Proces má za úkol zkontrolovat, zda je připojena záznamová paměť, zjistí to tím, že na zařízení je nahrán prázdný soubor s názvem *nemazat.dat*. Dále zajistí bezdrátové připojení k počítači, pokud je dostupný, děje se tak přes síťové rozhraní s přípojeným wifi přístupovým bodem. Poté zajistí čtení ze sběrnice CAN, vyhodnocení informace dle tabulky dostupných signálů a případné uložení na záznamové zařízení. Zaznamenaná data mají u sebe časovou značku, pro jednodušší analýzu dat.

Celé programy jsou umístěny na přiloženém médiu. Jsou tam dvě verze, první vytvořená pro ukázku systému a bezdrátovou komunikaci se záznamem dat, která byla použita při prezentacích vozu. Druhá verze je pro ukládání dat na záznamové zařízení ve formátu pro načtení do programu matlab. Uvedu zde kousek ze záznamu na soutěži v Německu.

[08.08 12:51:35]	advance cyl1	main inj 1	sec inj 1	T_vody[°]
otacky[RPM]	napeti [mV]	acc. correc.	of inj.[%]*10	klapka [%]*10

12:51:35:924351	38	33	33	44	12505	12544	0	165
12:51:36:024598	34	35	35	44	12505	13312	0	335
12:51:36:125011	35	41	41	44	12505	13056	-1	333
12:51:36:226836	38	37	37	44	12505	13056	-3	185
12:51:36:326958	41	33	33	44	12505	13056	-3	142
12:51:37:014291	51	14	14	44	12505	13312	-2	9
12:51:37:355401	44	14	14	44	12505	13312	-1	7
12:51:37:360713	42	15	15	44	12505	13056	-1	9
12:51:37:365368	46	20	20	44	12505	13056	0	40
12:51:37:704936	41	29	29	44	12505	12800	0	139
12:51:37:710084	32	39	39	44	12505	13312	0	396
12:51:38:050740	41	39	39	44	12505	13056	-10	160
12:51:38:391898	46	13	13	44	12505	13312	-3	7
12:51:38:732901	47	14	14	44	12505	13312	-1	9
12:51:38:737367	46	15	15	44	4613	13312	-1	9
12:51:38:741553	45	22	22	44	4613	13056	0	46
12:51:39:081991	36	38	38	44	4613	12800	0	323
12:51:39:424306	37	44	44	44	4613	13056	0	572
12:51:39:767514	48	27	27	44	4613	13056	-13	115
12:51:40:108500	53	14	14	44	4613	12800	-2	9
12:51:40:113220	54	18	18	44	4613	12800	-1	33
12:51:40:119318	49	21	21	44	4613	13056	-1	41
12:51:42:499589	33	39	39	44	4613	12800	0	382
12:51:42:505761	36	40	40	44	4613	13312	0	414
12:51:42:845377	35	43	43	44	4613	13056	-1	413

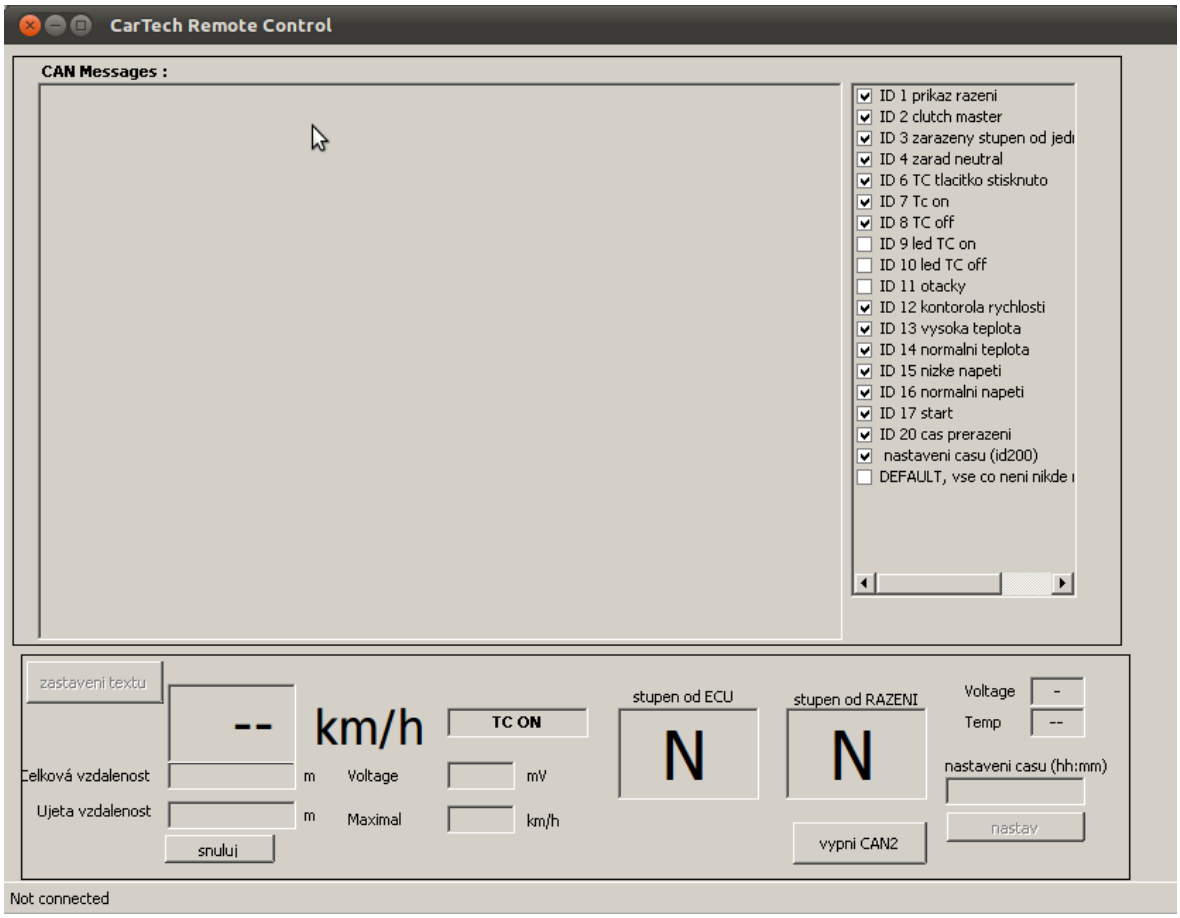
Z ukázky je vidět charakter zaznamenávaných dat a také je vidět, že byl vybrán soubor zaznamenávající hodnoty týkající se motoru. Jako první je čas zobrazující hodiny, minuty, sekundy s rozlišením na 6 desetinných míst pro rozeznání po sobě jdoucích záznamů. Hodnoty jsou mezi sebou odděleny tabulátory.

Při záznamu dat se zapisují hodnoty do 4 souborů, které jsou rozděleny dle charakteru informací. Jsou to soubory: can_draha.dat, can_ecu.dat, can_id0.dat a can_rychlost.dat. Názvy prozradí, jaká data jsou ukládána:

- can_draha.dat obsahuje ujetou vzdálenost
- can_ecu.dat obsahuje informace o motoru
- can_id0.dat obsahuje informace na první sběrnici CAN, kam chodí zprávy z jednotky řazení
- can_rychlost.dat obsahuje rychlost, kterou jednotlivá kola jela.

8.4.2 Program pro spojení s počítačem

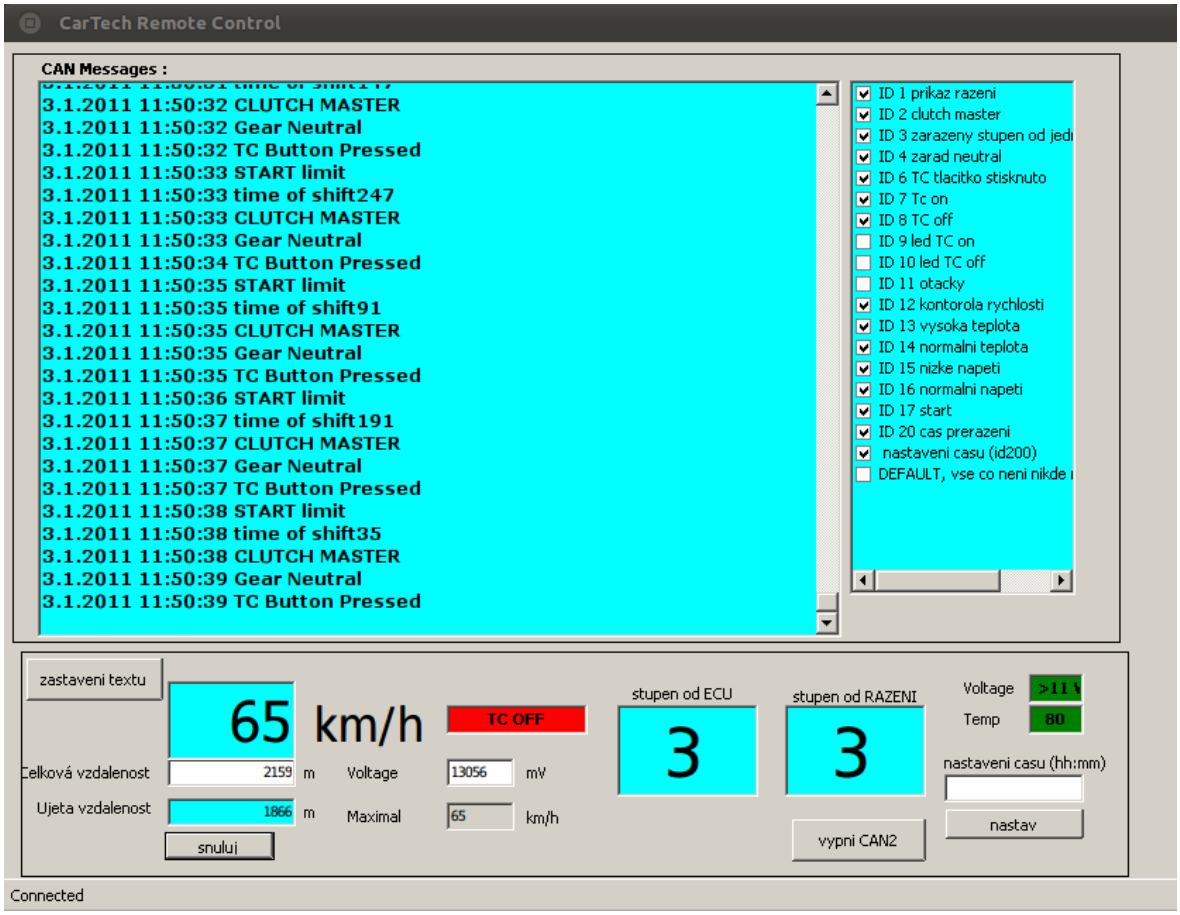
Při bezdrátovém přenosu byly posílány informace a aby je bylo kde zobrazit, tak jsem napsal program, který bude přes síťové rozhraní komunikovat s procesorovou deskou a zobrazoval přijatá data. Pro jeho vytvoření jsem použil programovací jazyk C#, jehož zdrojový kód je na přiloženém médiu.



Obrázek 8.3: Program pro síťové spojení s počítačem (odpojený stav)

Program je vidět na obrázku 8.3 a 8.4, kde je dominantní veliké pole pro výpis událostí. V pravé části je možné si odfiltrovat zprávy. V dolní části jsou zobrazované informace o okamžité rychlosti, o ujeté vzdálenosti, o palubním napětí, o teplotě, o maximální rychlosti, rychlostní stupeň zařazený podle řazení, podle řídící jednotky. Dále je možnost nastavit čas pro záznam dat, kdyby bylo potřeba zajistit nějakou znatelnou značku v datech.

Aby program správně fungoval, je potřeba mít nastavenou IP adresu na 192.168.1.4.a procesorový modul na IP adresu 192.168.1.1, nebo si nechat libovolnou a akorát ji změnit v programu.

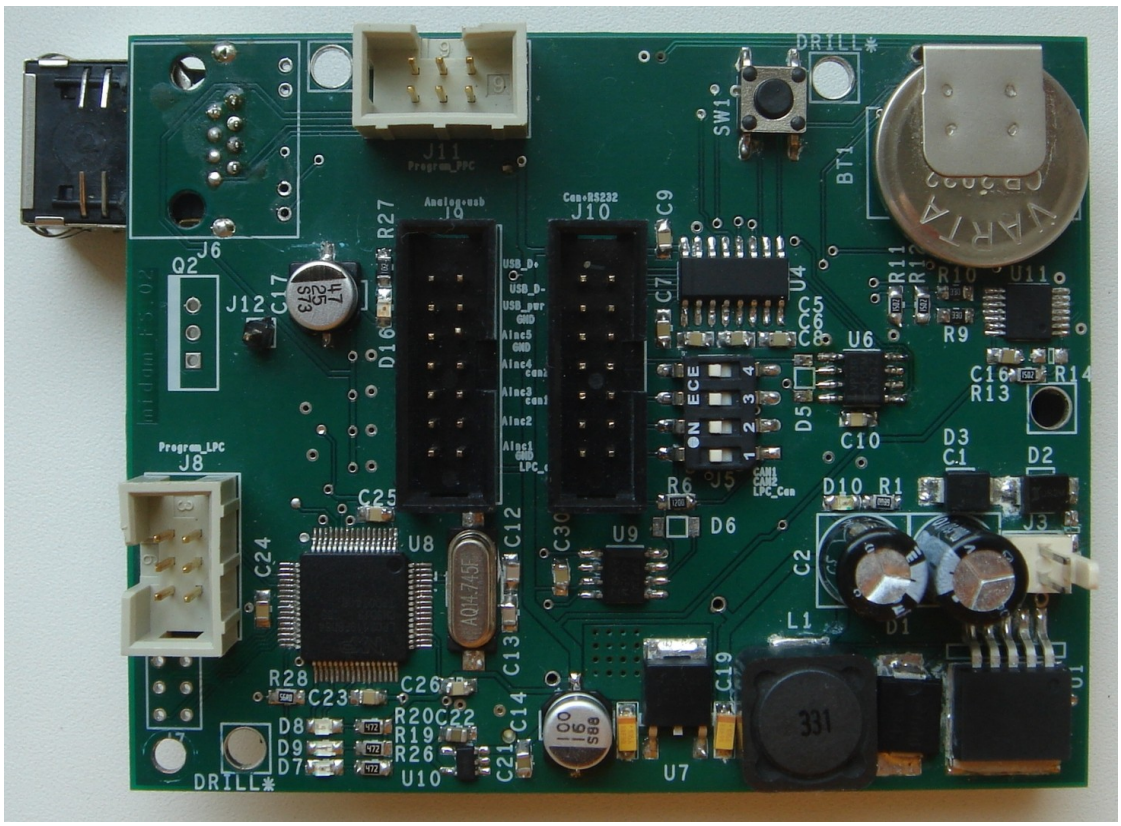


Obrázek 8.4: Program pro síťové spojení s počítačem (spojený stav)

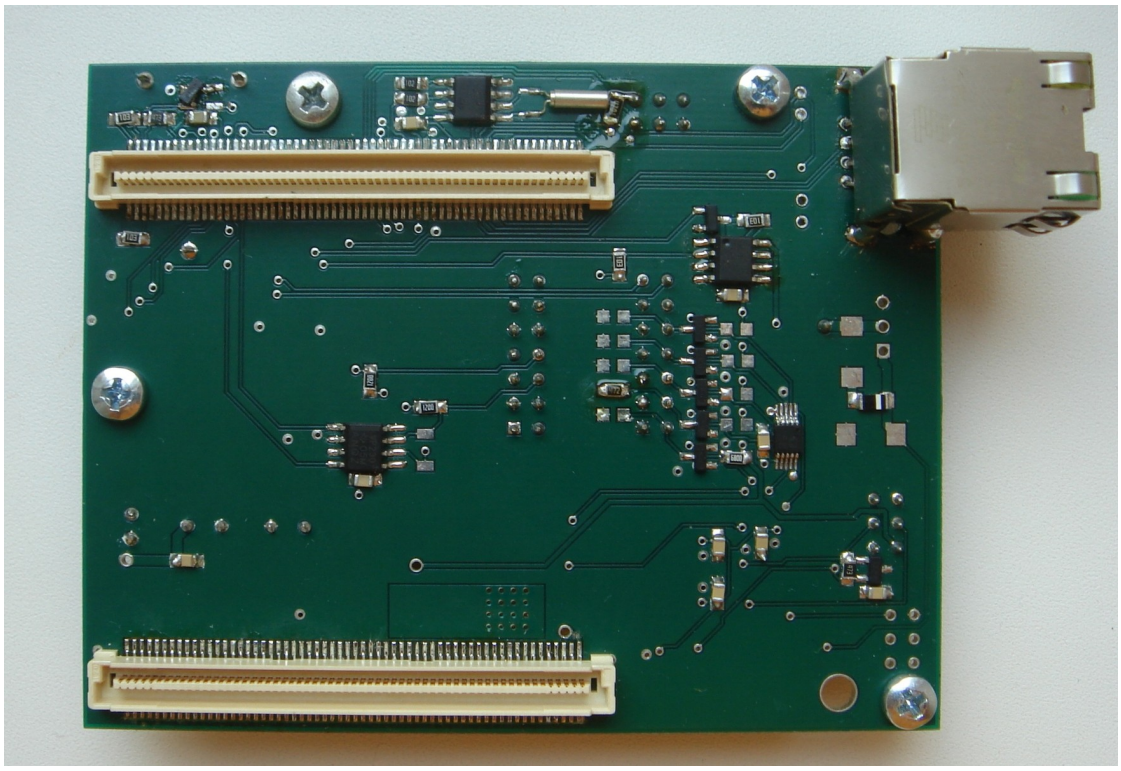
9 Realizace

V průběhu sestavování desky plošných spojů jsem se dozvěděl, že chybí finance na nákup gps modulu a senzoru natočení volantu. Tyto senzory jsou postradatelné a nejsou stěžejní pro analýzu chování vozu. Senzor natočení volantu jsem změnil na více otáčkový potenciometr, ten je finančně mnohem dostupnější. Po delší době nebyly finance na senzory tlumičů, pro ně se ovšem později našel sponzor. Který je ale bohužel nedodal, protože je neměl skladem. V poslední řadě se pro nedostatek času při výrobě vozu nepřivařilo uchycení pro senzor natočení volantu, tím došlo k nevyužití analogových vstupů na desce.

Při oživování plošného spoje jsem objevil pár nedostatků. Jedním z nich byla absence pull-up rezistorů u pinu procesorového modulu RXD a u pinu procesoru LPC2119 LPC_boot. Bez těchto rezistorů ani jeden z procesorů nenaběhl. Po odstranění chyby jsem nemohl stále načíst procesorový modul. Odhalil jsem špatný typ resetovacího obvodu. Byl mi zaslán obvod s vyšší hranicí, kdy jde obvod do resetu, proto byl modul stále v resetu. Po těchto změnách celá deska korektně naběhla. Poslední nedostatek jsem nalezl v síťovém konektoru, u kterého jsem objednal nesprávný kus, neměl transformátor. Bohužel jsem nenašel žádný pinově kompatibilní, proto nový konektor vyčnívá z desky. Jak vypadá deska ve skutečnosti lze vidět na obrázku 9.1 a 9.2.

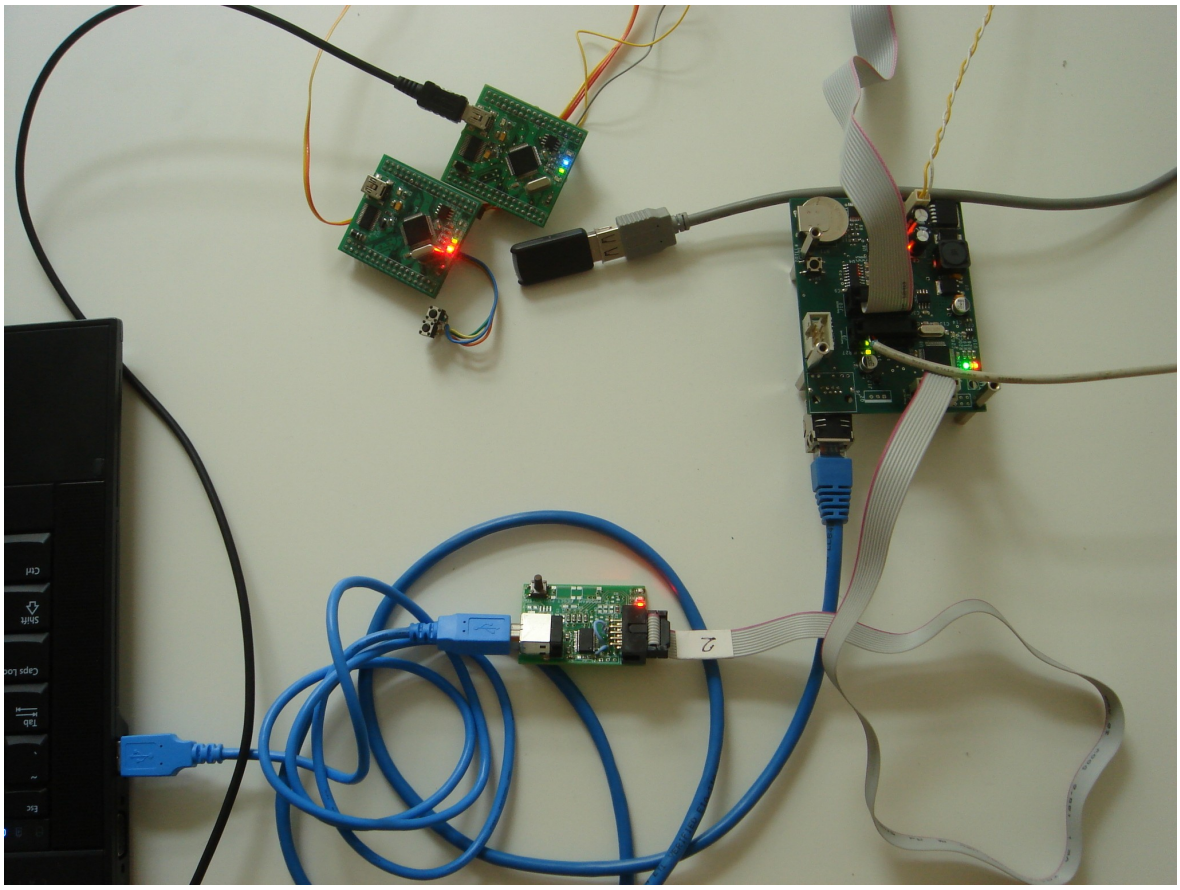


Obrázek 9.1: Svrchní strana desky


Obrázek 9.2: Spodní strana desky

Po hardwarových úpravách jsem odzkoušel procesory nahráním jednoduchých programů. U procesoru LPC2119 to byl program, který střídavě rozsvěcel LED. Procesorová deska posílala zprávu na jeden kanál sběrnice CAN a z druhé jej četla, sběrnice byly v konektoru propojené.

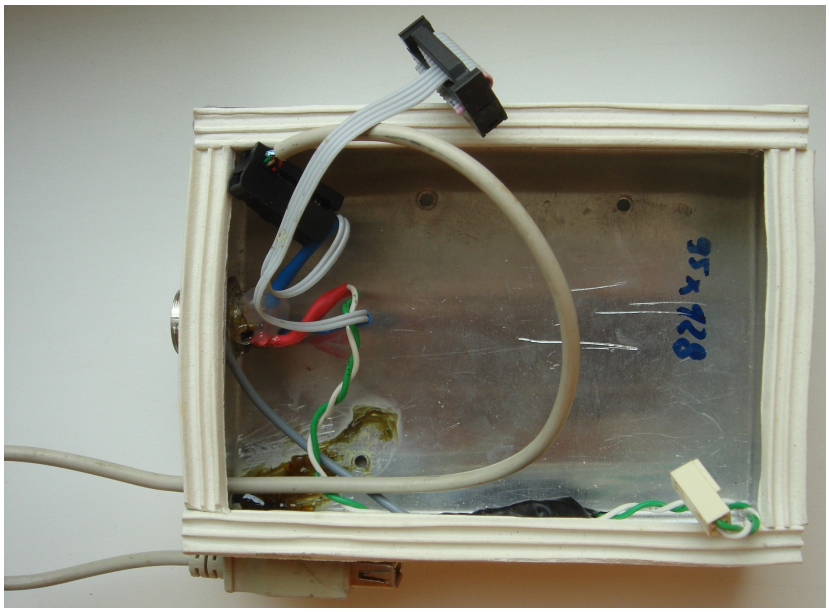
Na testování a dolaďování programu jsem používal dvě samostatné desky s LPC. Tyto moduly se jmenují LpcEurobot [24], byly vyvinuty pro robota na soutěž Eurobot. Do nich jsem nahrál program, který posílá po sběrnici CAN zprávy stejných formátů, jako byly zprávy ve vozu. Dva moduly z toho důvodu, abych simuloval situaci dvou sběrnic a čtení z nich zároveň. Ukázka testovací soustavy je na obrázku 9.3.



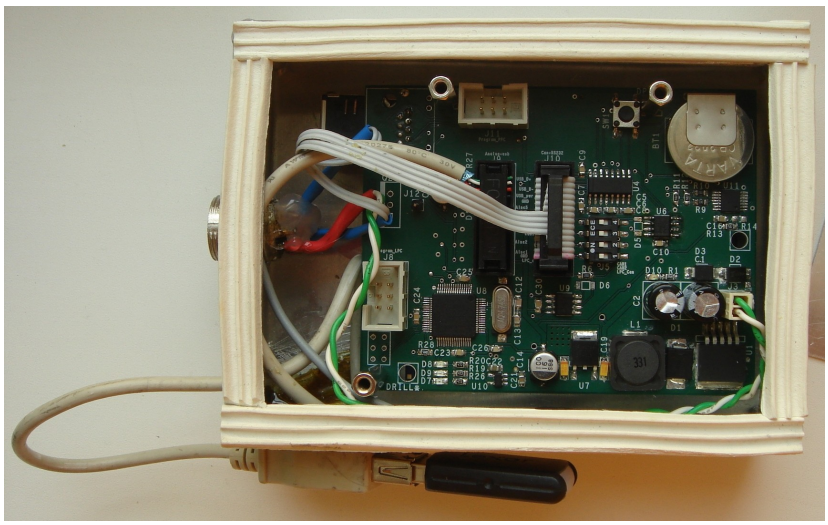
Obrázek 9.3: Testování a dolaďování záznamové desky

Pro umístění zařízení ve voze se vybralo místo v levém tunelu auta. Byl zde nevyužitý prostor. Toto místo je chráněné proti přímému dešti a slunečnímu záření, ale je zde zvýšený výskyt odletujícího štěrku od pneumatik vozu, v případě deště se sem dostane i voda. Z těchto důvodů bylo potřeba umístit záznamové zařízení do ochranné boxu. Pro tento účel jsem vyrobil ochranný box z hliníkového plechu a namontoval jsem na něj

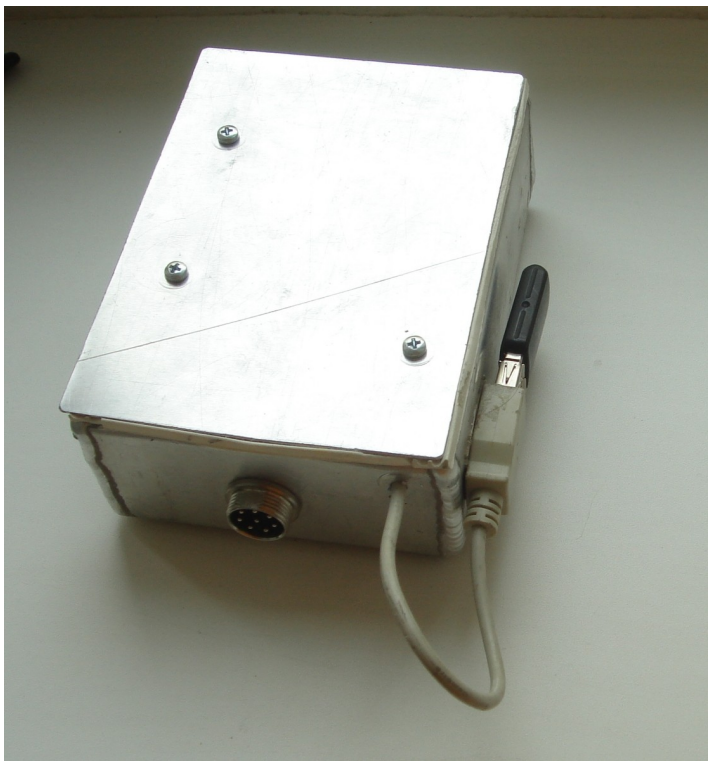
konektor pro připojení kabeláže vozu. Na obrázcích 9.4, 9.5, 9.6 a 9.7 je vidět provedení ochranného boxu i s umístěním desky. Ochranný box je utěsněna pomocí těsnění, které se používá na okenní rám. Úložné zařízení je umístěno mimo ochranný box, aby bylo jednoduše dostupné. Celý box je připevněn k vozu pomocí suchých zipů, které jsou nadimenzovány, tak aby ho udržely.



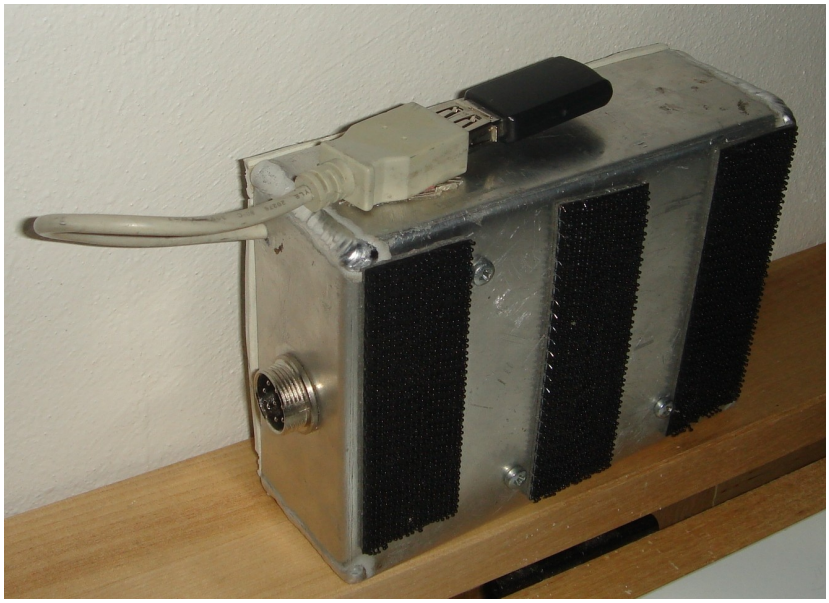
Obrázek 9.4: Ochranný box prázdný



Obrázek 9.5: Ochranný box s modulem



Obrázek 9.6: Ochranný box ze shora



Obrázek 9.7: Ochranný box zespodu

10 Zhodnocení

V této diplomové práci jsem se zabýval návrhem a realizací modulu pro shromáždění a záznam dat z vozu FS02. V rámci této práce jsem navrhl hardware pro záznam dat s názvem midam_FS.02. Ten obsahuje procesor LPC a procesorový modul MIDAM od firmy Mikroklima [18]. Procesor je typu ARM7 a na procesorovém modulu je umístěn procesor PowerPC. Výrobní data jsem umístil na přiložené DVD. U tohoto bodu jsem odhalil drobné nedostatky, které jsou popsány v kapitole realizace. Kvůli těmto nedostatkům není třeba předělávat celý hardware, postačí pouze pozměnit síťový konektor a přidat dva rezistory.

Médium pro ukládání dat je flash disk připojitelný přes USB konektor, s nímž je spojen s navrženou deskou. S flash diskem je propojený procesorový modul, který na něj ukládá data. Deska tyto informace získává ze sběrnice CAN, po které komunikují jednotky umístěné v automobilu. Pro ukládání potřebných dat jsem napsal program v jazyce C, ten je také umístěn na přiloženém DVD.

Pro demonstraci vznikl také program napsaný v jazyce C#, který je spustitelný na uživatelském počítači. Tato aplikace spojuje počítač a procesorový modul přes síťové rozhraní. Pomocí wifi je zpřístupněná možnost bezdrátového přenosu informací. Takto byl také využíván, zobrazí různé informace o stavu vozu.

Reference:

[1] *ISA-RACING : datalogging* [online]. c2010 [cit. 2010-01-04]. Dostupný z WWW: <http://www.isa-racing.com/index.php/cat/c539_STACK-ST800-Datenaufzeichnung.html/XTCsid/b0f3526a377711fd720e034ef879a7c4>.

[2] Motosport-bosch : data logger C sport. *Datasheet* [online]. 2009 [cit. 2010-01-04]. Dostupný z WWW: <http://www.bosch-motorsport.de/content/language2/html/2962.htm>.

[3] MoTeC : *ACL* [online]. c2010 [cit. 2010-01-04]. Dostupný z WWW: <http://www.siliconchip.com.au/cms/A_107099/article.html>.

[4] TOSHIKAZU, Ina, et al. 360-degree Rotation Angle Sensor Consisting of MRE Sensors with a Membrane Coil. *SOKEN* [online]. 2005 [cit. 2010-01-04]. Dostupný z WWW: <http://www.amaa.de/previous/amaa_2005/presentations2005/01-02-03_Toshikazu-Ina.pdf>.

[5] *AMCI* [online]. c2010 [cit. 2011-01-02]. Resolver. Dostupné z WWW: <http://www.amci.com/tutorials/tutorials-what-is-resolver.asp>

[6] Sas, Geert (Meerhout, BE), Maes, Mario (Zutendaal, BE), Lauwerys, Xavier (Landen, BE), Mermans, Pol (Mielen-boven-Aalst, BE), Chiaradia, David (Alken, BE). Shock absorber with integrated displacement sensor. [online]. 2006 [cit. 2010-01-04]. Dostupný z WWW: <http://www.freepatentsonline.com/y2006/0011426.html>.

[7] Autospeed*: technical features* [online]. 2007 [cit. 2010-01-04]. Dostupný z WWW: <http://autospeed.com/cms/title_MoTeCs-Race-Car-Data-Logging-Part-1/A_108254/article.html>.

[8] Mirco-Epsilon UK Ltd: produc new [online]. 2009 [cit. 2010-01-04]. Dostupný z WWW: <http://www.sensorland.com/AppPage104.html>.

[9] Ford-club [online]. 2010 [cit. 2010-12-18]. Systémy stability. Dostupné z WWW: <http://www.ford-club.cz/2010/02/systemy-stability/>

[10] Hall effect. In *Wikipedia : the free encyclopedia* [online]. St. Petersburg (Florida) : Wikipedia Foundation, 2004, last modified on 2010-12-1 [cit. 2010-12-19]. Dostupné z WWW: <http://en.wikipedia.org/wiki/Hall_effect>

[11] *EETimes* [online]. [cit. 2010-12-19]. Exploring optical and magnetic sensors. Dostupné z WWW: <http://www.eetimes.com/design/embedded/4024586/Exploring-optical-and-magnetic-sensors>

[12] Press release*:Bourns Announces Customized Steering Angle Sensors Based on AMR Technology* [online]. 2009 [cit. 2010-01-04]. Dostupný z WWW: <http://www.bourns.com/PressReleases.aspx?name=090901_steering_angle>.

[13] SOBOTKA, Jan. *Traction Control System for the formula CTU CarTech* [online]. Praha : 2010. 73 s. Diplomová práce. ČVUT v Praze. Dostupné z WWW: <https://dip.felk.cvut.cz/browse/details.php?f=F3&d=K13136&y=2010&a=sobotja2&t=dipl>

[14] Navilock *: Produkte* [online]. c2010 [cit. 2010-01-04]. Dostupný z WWW: <http://www.navilock.de/produkte/>.

[15] *HW wiki:* [online]. 2008 [cit. 2010-12-11]. MIDAM MPC5200. Dostupné z WWW: <http://rtime.felk.cvut.cz/hw/images/b/b0/SHARK_V1_1_DOC_CZ.pdf>

[16] Freescale *: MPC5200* [online]. c2004-2010 [cit. 2010-01-04]. Dostupný z WWW: <http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=MPC5200>.

[17] Freescale Semiconductor, Inc [online]. 2006-03 [cit. 2010-12-12]. MPC5200 Hardware Specification. Dostupné z WWW: <http://cache.freescale.com/files/32bit/doc/data_sheet/MPC5200.pdf>

[18] *Midam control system* [online]. 2008 [cit. 2010-12-06]. MIDAM SHARK. Dostupné z WWW: <http://www.midam.cz/products/MIDAM-SHARK.html>

[19] *LPC2109/2119/2129* [online]. [s.l.] : NXP B.V., 2007 [cit. 2011-01-01]. Dostupné z WWW: <http://www.nxp.com/documents/data_sheet/LPC2109_2119_2129.pdf>

[20] *Datasheet LM2575,* National Semiconductor, 2004-08.

[21] *Měření a zpracování dat v dopravě* [online]. 2010 [cit. 2010-12-27]. CAN bus. Dostupné z WWW: <http://euler.fd.cvut.cz/predmety/pmzd/data/MZD_05_can_bus_2010.pdf>

[22] ČAREK, Lukáš. *ŘÍDICÍ SYSTÉM PRO REGULACI TEPLOTY*. Praha, 2008. 111 s. Diplomová práce. ČVUT v Praze. Dostupné z WWW: <http://support.dce.felk.cvut.cz/mediawiki/images/1/10/Dp_2008_carek_lukas.pdf>

[23] *Katedra řídící techniky* [online]. [cit. 2010-12-30]. OMK git. Dostupné z WWW: <http://rtime.felk.cvut.cz/gitweb/omk.git>

[24] KUBIAS, Jiří. *Hardware robota pro soutěž Eurobot*. Liberec, 2010. 85 s. Diplomová práce. ČVUT v Praze.

Přílohy

A Zapojení konektoru procesorové desky MPC5200

Konektor 1		
číslo pinu	funkce/signál	skupina
pinu		
1	+3V3	Power
2	GND	Power
3	-	-
4	-	-
5	-	-
6	-	-
7	-	-
8	-	-
9	-	-
10	GND	Power
11	-	-
12	-	-
13	+3V3	Power
14	-	-
15	-	-
16	-	-
17	-	-
18	GND	Power
19	-	-
20	-	-
21	-	-
22	-	-
23	IRQ3	Interrupt
24	-	-
25	+3V3	Power
26	GND	Power

číslo pinu	funkce/signál	skupina
27	IRQ1	Interrupt
28	IRQ2	Interrupt
29	PCI_STOP#	PCI
30	IRQ0	Interrupt
31	PCI_RESET#	PCI
32	PCI_TRDY#	PCI
33	PCI_PERR#	PCI
34	GND	Power
35	PCI_IRDY#	PCI
36	PCI_SERR#	PCI
37	+3V3	Power
38	PCI_REQ#	PCI
39	PCI_GNT#	PCI
40	PCI_PAR	PCI
41	PCI_DEVSEL#	PCI
42	GND	Power
43	PCI_CBE_3#	PCI
44	PCI_ID_SEL	PCI
45	PCI_CBE_2#	PCI
46	PCI_FRAME#	PCI
47	PCI_CBE_1#	PCI
48	PCI_CLOCK	PCI
49	+3V3	Power
50	GND	Power
51	PCI_CBE_0#	PCI
52	EXT_AD_30	PCI
53	EXT_AD_31	PCI
54	EXT_AD_28	PCI
55	EXT_AD_29	PCI
56	EXT_AD_26	PCI
57	EXT_AD_27	PCI
58	GND	Power
59	EXT_AD_25	PCI
60	EXT_AD_24	PCI

číslo pinu	funkce/signál	skupina
61	+3V3	Power
62	EXT_AD_22	PCI
63	EXT_AD_23	PCI
64	EXT_AD_20	PCI
65	EXT_AD_21	PCI
66	GND	Power
67	EXT_AD_19	PCI
68	EXT_AD_18	PCI
69	EXT_AD_17	PCI
70	EXT_AD_16	PCI
71	EXT_AD_15	PCI
72	EXT_AD_14	PCI
73	+3V3	Power
74	GND	Power
75	EXT_AD_13	PCI
76	EXT_AD_12	PCI
77	EXT_AD_11	PCI
78	EXT_AD_10	PCI
79	EXT_AD_9	PCI
80	EXT_AD_8	PCI
81	EXT_AD_7	PCI
82	GND	Power
83	EXT_AD_5	PCI
84	EXT_AD_6	PCI
85	+3V3	Power
86	EXT_AD_4	PCI
87	EXT_AD_3	PCI
88	EXT_AD_2	PCI
89	EXT_AD_1	PCI
90	GND	Power
91	LP_R/W#/CFG_3	LP
92	EXT_AD_0	PCI
93	LP_ALE#/CFG_4	LP
94	LP_TS#/CFG_5	LP

číslo pinu	funkce/signál	skupina
95	LP_ACK	LP
96	LP_OE#	LP
97	+3V3	Power
98	GND	Power
99	-	-
100	-	-
101	-	-
102	-	-
103	-	-
104	-	-
105	-	-
106	GND	Power
107	-	-
108	OUT_CS5	CS
109	+3V3	Power
110	OUT_CS4	CS
111	OUT_CS3	CS
112	OUT_CS2	CS
113	OUT_CS1	CS
114	GND	Power
115	-	-
116	OUT_CS0	CS
117	-	-
118	-	-
119	+3V3	Power
120	GND	Power

Konektor 2		
číslo pinu	funkce/signál	skupina
pinu		
121	+3V3	Power
122	GND	Power
123	HRESET#	Reset

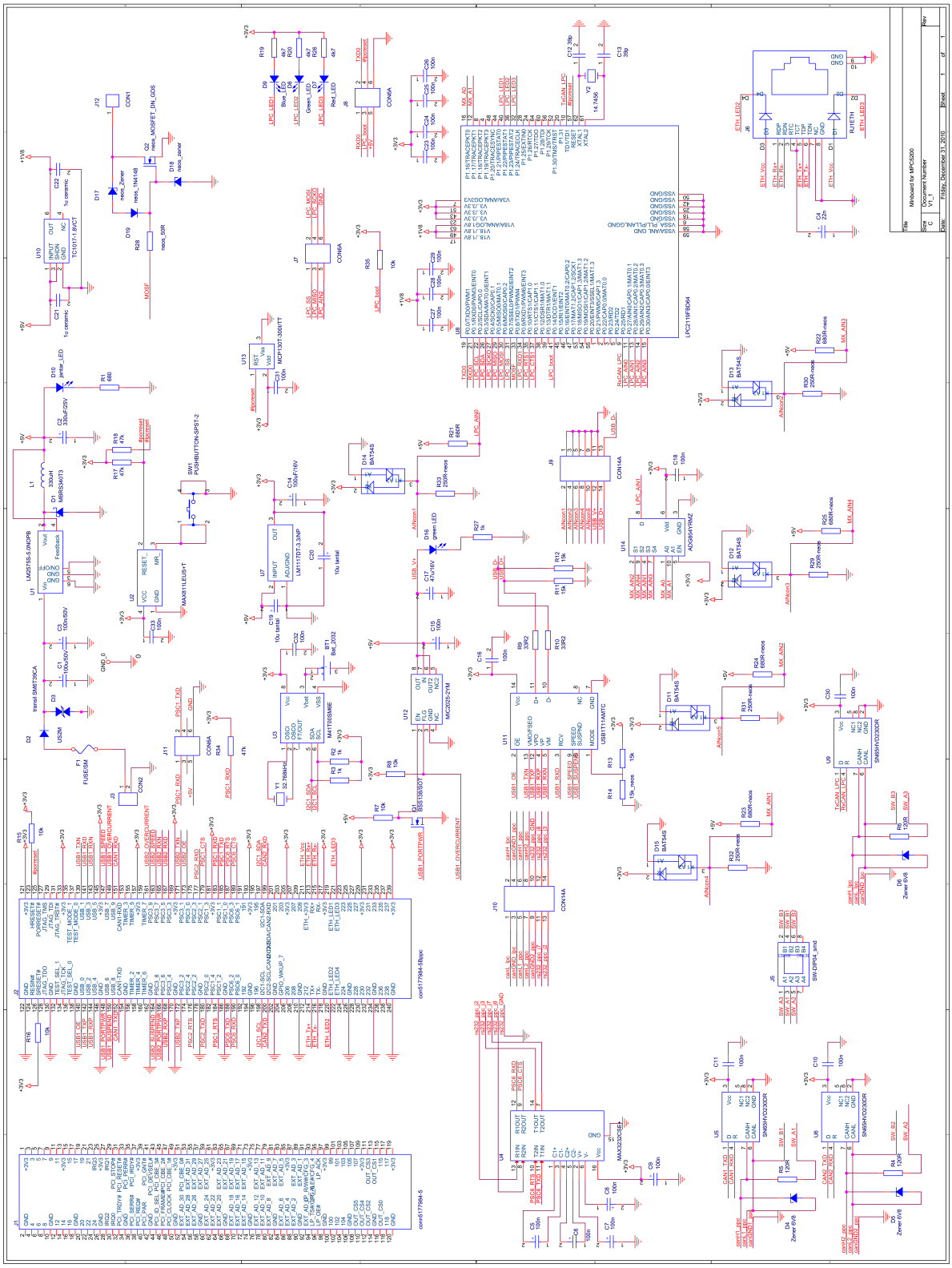
číslo pinu	funkce/signál	skupina
124	RESIN#	Reset
125	PORRESET#	Reset
126	SRESET#	Reset
127	JTAG_TMS	JTAG
128	JTAG_TDO	JTAG
129	JTAG_TDI	JTAG
130	GND	Power
131	JTAG_TRST#	JTAG
132	TEST_SEL_1	JTAG
133	+3V3	Power
134	JTAG_TCK	JTAG
135	TEST_MODE_1	JTAG
136	TEST_SEL_0	JTAG
137	TEST_MODE_0	JTAG
138	GND	Power
139	USB_1	USB
140	USB_0	USB
141	USB_3	USB
142	USB_2	USB
143	USB_5	USB
144	USB_4	USB
145	+3V3	Power
146	GND	Power
147	USB_7	USB
148	USB_6	USB
149	USB_9	USB
150	USB_8	USB
151	CAN1-RXD	CAN
152	CAN1-TXD	CAN
153	TIMER_3	TIMER
154	GND	Power
155	TIMER_5	TIMER
156	TIMER_2	TIMER
157	+3V3	Power

číslo pinu	funkce/signál	skupina
158	TIMER_4	TIMER
159	TIMER_7	TIMER
160	TIMER_6	TIMER
161	PSC3_9	PSC3
162	GND	Power
163	PSC3_7	PSC3
164	PSC3_8	PSC3
165	PSC3_5	PSC3
166	PSC3_6	PSC3
167	PSC3_3	PSC3
168	PSC3_4	PSC3
169	+3V3	Power
170	GND	Power
171	PSC3_1	PSC3
172	PSC3_2	PSC3
173	PSC3_0	PSC3
174	PSC2_4	PSC2
175	PSC2_3	PSC2
176	PSC2_2	PSC2
177	PSC2_1	PSC2
178	GND	Power
179	PSC1_3	PSC1
180	PSC2_0	PSC2
181	+3V3	Power
182	PSC1_4	PSC1
183	PSC1_1	PSC1
184	PSC1_2	PSC1
185	PSC1_0	PSC1
186	GND	Power
187	PSC6_3	PSC6
188	PSC6_2	PSC6
189	PSC6_1	PSC6
190	PSC6_0	PSC6
191	-	-

číslo pinu	funkce/signál	skupina
192	-	-
193	+3V3	Power
194	GND	Power
195	-	-
196	-	-
197	I2C1-SDA	I2C
198	I2C1-SCL	I2C
199	CAN2-RXD	CAN
200	CAN2-TXD	CAN
201	-	-
202	GND	Power
203	-	-
204	GPIO_WKUP_7	GPIO
205	+3V3	Power
206	-	-
207	-	-
208	-	-
209	-	-
210	GND	Power
211	ETH_+3V3	Ethernet
212	-	-
213	RX+	Ethernet
214	TX+	Ethernet
215	RX-	Ethernet
216	TX-	Ethernet
217	+3V3	Power
218	GND	Power
219	ETH_LED1	Ethernet
220	ETH_LED2	Ethernet
221	ETH_LED3	Ethernet
222	ETH_LED4	Ethernet
223	-	-
224	-	-
225	-	-

číslo pinu	funkce/signál	skupina
226	GND	Power
227	-	-
228	-	-
229	+3V3	Power
230	-	-
231	-	-
232	-	-
233	-	-
234	GND	Power
235	-	-
236	-	-
237	-	-
238	-	-
239	+3V3	Power
240	GND	Power

B Schéma desky MIDAM_FS02



C Tabulka součástek použitých ve schématu pro desku MIDAM_FS02

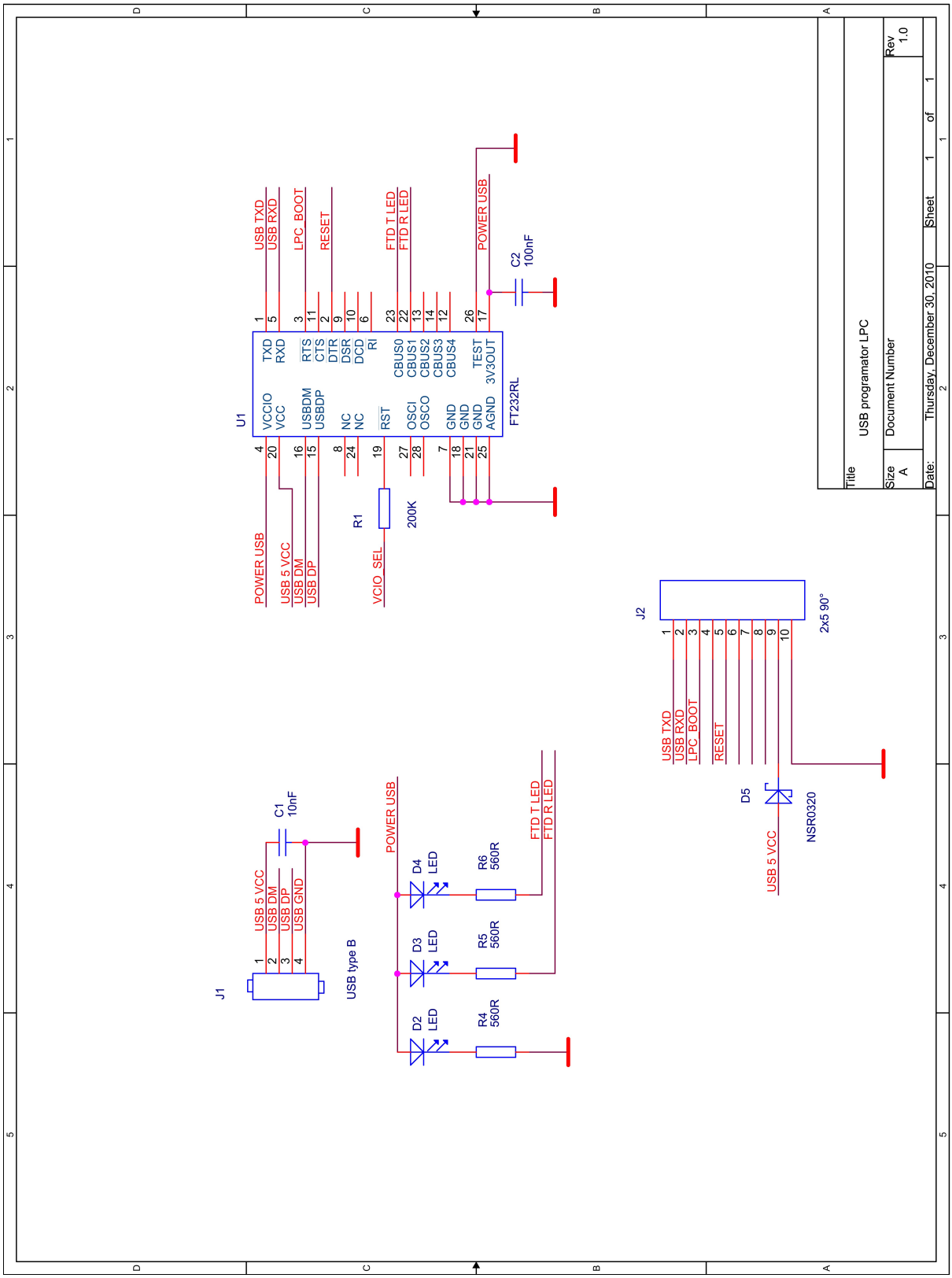
Komponenta	Pouzdro + popis	Hodnota
BT1	baterie CR2032 s pouzdrem	CR2032
C1	8 x 11mm	100uF/50V
C2	8 x 11mm	330uF/25V
C3	0805	100nF/50V
C4	0805	22nF
C5	0805	100nF
C6	0805	100nF
C7	00805	100nF
C8	0805	100nF
C9	0805	100nF
C10	0805	100nF
C11	0805	100nF
C12	0805	39pF
C13	0805	39pF
C14	kondenzátor smd 6,3x5,5mm	100uF/16V
C15	0805	100nF
C16	0805	100nF
C17	kondenzátor smd 6,3x5,5mm	47uF/16V
C18	0805	100nF
C19	1206, tantalový	10uF
C20	1206, tantalový	10uF
C21	00805, keramický	1uF
C22	00805, keramický	1uF
C23	0805	100nF
C24	0805	100nF
C25	0805	100nF
C26	0805	100nF
C27	0805	100nF
C28	0805	100nF
C29	0805	100nF
C30	0805	100nF

Komponenta	Pouzdro + popis	Hodnota
C31	0805	100nF
C32	0805	100nF
C33	0805	100nF
D1	SMC	MBRS340T3
D2	SMB	US2M
D3	SMB	SM6T39CA
D4	SOD-80C	Zener 6V8
D5	SOD-80C	Zener 6V8
D6	SOD-80C	Zener 6V8
D7	0805	Red_LED
D8	0805	Green_LED
D9	0805	Blue_LED
D10	0805	jantar_LED
D11	sot-23	BAT54S
D12	sot-23	BAT54S
D13	sot-23	BAT54S
D14	sot-23	BAT54S
D15	sot-23	BAT54S
D16	0805	green LED
D17	SMB, zenerova dioda	
D18	SMB, zenerova dioda	
D19	1206	1N4148
F1	0805	BSMD00805
J1	5177984-5	5177984-5
J2	5177984-5	5177984-5
J3	Napájecí konektor	
J5	4 x přepínač	
J6	síťový konektor	
J7	pinová lišta 2x3	
J8	konektor přímý, 6 pinů	
J9	konektor přímý,14 pinů	
J10	konektor přímý,14 pinů	
J11	konektor přímý, 6 pinů	
J12	pin	

Komponenta	Pouzdro + popis	Hodnota
L1	de1207	330uH
Q1	sot-23	BSS138
Q2	to-220ab	STP36NF06L
R1	0805	680R
R2	0805	1k
R3	0805	1k
R4	0805	120R
R5	0805	120R
R6	0805	120R
R7	0805	10k
R8	0805	10k
R9	0805	33R2
R10	0805	33R2
R11	0805	15k
R12	0805	15k
R13	0805	15k
R14	0805	15k
R15	0805	10k
R16	0805	10k
R17	0805	47k
R18	0805	47k
R19	0805	4k7
R20	0805	4k7
R21	0805	680R
R22	0805	680R
R23	0805	680R
R24	0805	680R
R25	0805	680R
R26	0805	4k7
R27	0805	1k
R28	0805	50R
R29	0805	250R
R30	0805	250R
R31	0805	250R

Komponenta	Pouzdro + popis	Hodnota
R32	0805	250R
R33	0805	250R
R34	0805	47k
R35	00805	10k
SW1	Mikrotlačítko 6 mm x 6 mm	
U1	TO263	LM2575S-5.0NOPB
U2	sot143	MAX811LEUS+T
U3	so8	M41T00SM6E
U4	soic16	MAX3232CSE+
U5	so8	SN65HVD230DR
U6	so8	SN65HVD230DR
U7	to252	LM1117DT-3.3/NP
U8	lqfp64	LPC2119FBD64
U9	so8	SN65HVD230DR
U10	sot23-5	TC1017-1.8VCT
U11	tssop14	USB1T11AMTC
U12	so8	MIC2025-2YM
U13	sot-23	MCP130T-300I/TT
U14	msop10	ADG804YRMZ
Y1	Krystal 6x2smd	32.768kHz
Y2	HC49SM	14.7456

D Schéma zapojení programátoru



E Obsah DVD

K této práci je přiloženo DVD, na kterém jsou uloženy zdrojové kódy a výrobní data.

- Výrobní data pro desku MIDAM_FS02
- Zdrojové kódy programů napsaných pro tuto práci
- Přílohy této práce
- Tato práce