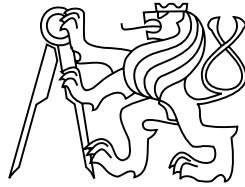


České vysoké učení technické v Praze

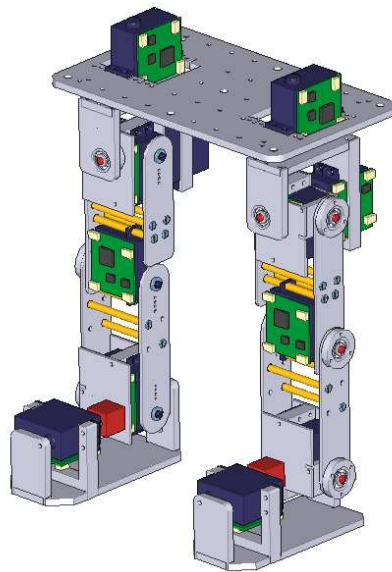
Fakulta elektrotechnická

katedra řídicí techniky



Kráčející robot

diplomová práce



vytvořil Marek Peca
pod vedením Ing. Michala Sojky

Praha 2008

Katedra řídicí techniky

Školní rok: 2005/2006

ZADÁNÍ DIPLOMOVÉ PRÁCE

Student: Marek P e c a

Obor: Technická kybernetika

Název tématu: Kráčejíci robot

Zásady pro vypracování:

1. Seznamte se s principy tvorby a řízení kráčejících (dvounohých) robotů.
2. Navrhněte a realizujte malé servomotory a jejich řídicí elektroniku. Serva budou ovládaná přes sběrnici CAN. Serva využijte jako akční členy pro pohyb robotu.
3. Navrhněte a sestrojte mechaniku takového robotu.
4. Zvolte vhodný hardware pro hlavní řídicí jednotku robotu a naprogramujte jednoduché řízení pohybu robotu.
5. Naprogramujte interface pro řízení robotu z Matlabu.

Seznam odborné literatury: Dodá vedoucí práce.

Vedoucí diplomové práce: Ing. Michal Sojka

Termín zadání diplomové práce: zimní semestr 2005/2006

Termín odevzdání diplomové práce: leden 2007

prof. Ing. Michael Šebek, DrSc.
vedoucí katedry



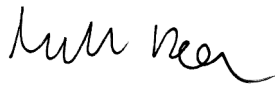
prof. Ing. Zbyněk Škvor, CSc.
děkan

Prohlášení

Práci jsem vypracoval samostatně, za cenné rady vděčím kolegům a učitelům. Literatura, z níž jsem čerpal, je uvedena v seznamu.

(Oproti původnímu zadání práce mi bylo vedoucím a Doc. Zdeňkem Hanzálkem řečeno, že nemám řešit rozhraní k programu Matlab. Tím úkolem se totiž má zabývat někdo jiný na pracovišti.)

V Praze dne 1. 1. 2008

A handwritten signature in black ink, appearing to read 'Marek Peca', written in a cursive style.

Marek Peca

Anotace

Práce představuje stavbu dvounohého kráčejícího robota, jeho elektroniku a základní software. Robot dosáhl základního vytčeného cíle, statické chůze s dlouhodobým stojem na obou nohou a pevnou trajektorií v kloubových souřadnicích. Robot se dvěma nohama, bez ostatních částí těla, je schopen chodit podle pevné, ručně vytvořené, statické trajektorie, řízen jednoduchými proporcionálními SISO regulátory. Navržen je tak, aby bylo možné použít i MIMO regulátory, pružnou trajektorii nebo dynamickou chůzi. Servomotory robota tvoří distribuovaný systém, řízený přes průmyslovou sběrnici CAN. V práci jsem zdůraznil podstatné body návrhu z hlediska mechanické konstrukce a kinematiky, proudového kaskádního řízení motorů a časování.

Annotation

Construction of a biped walking robot, its hardware, basic software and control design are being presented in this thesis. Primary goal achieved is a static walking with non-instantaneous double support phase and fixed trajectory in joint coordinates. The robot with two legs and no upper body is capable to walk with fixed, manually created, static trajectory using simple SISO proportional controller, yet it is extendable to use MIMO controllers, flexible trajectory, and dynamic gait. Distributed servo motor control over a CAN fieldbus is used. Important points in construction and kinematics, motor current cascaded control and fieldbus timing are emphasized.

Poděkování

Poděkování patří zejména mým blízkým za jejich ∞ trpělivost. Dále jsem velmi vděčen profesorům a učitelům za poučení a vedení, zejména Ing. Michalu Sojkovi, Ing. Pavlu Píšovi a Doc. Zdeňku Hanzálkovi.

věnováno naší sličné asistentce pro vědu a výzkum

鈴雫

Obsah

1. Úvod	3
2. Dvounohá chůze	4
2.1. Definice	4
2.2. Trajektorie chůze	4
2.2.1. Definice	4
2.2.2. Statická a dynamická chůze	5
2.2.3. Pevná a pružná trajektorie	5
2.2.4. Tvorba trajektorie	6
3. Mechanická konstrukce	7
3.1. Konstrukční řešení	7
3.2. Popis sestavení	8
4. Elektronika	10
4.1. Počítačová síť	10
4.2. Deska SPEJBL-ARM	11
4.3. Elektronika servomotoru	12
4.3.1. Původní elektronika	12
4.3.2. Vlastní elektronika	12
4.3.3. Přetrvávající chyby v návrhu	13
5. Software	14
5.1. Provoz na síti a časování	14
5.2. Programování SPEJBL-ARM	15
5.3. Ovládací program	16
5.3.1. Popis programu CerveauX	16
6. Zpětnovazební řízení	19
6.1. Rozdíly SISO a MIMO řízení	19
6.2. P regulátor	20
6.2.1. Rozbor	20
6.2.2. Realizace	22
6.2.3. Výsledky	22
6.3. Kaskádní PID-PI regulátor	23
6.3.1. Proudový PI regulátor	24
6.3.1.1. Filtrace PWM a převzorkování	25
6.3.1.2. Přenos decimovaného systému	26
6.3.1.2.1. Algoritmus decimace přenosu	29
6.3.1.3. Rozbor řízeného systému	30
6.3.1.4. Návrh regulátoru	31
6.3.1.5. Realizace	32
6.3.2. Polohový PID regulátor	34
6.3.2.1. Rozbor	34
6.3.2.2. Identifikace řízeného podsystemu	35
6.3.2.3. Návrh regulátoru	39
6.3.2.4. Realizace	40
6.3.3. Výsledky	41
6.3.3.1. Stabilita regulátoru uvnitř robota	43

6.3.3.2. Rušivé kmitání	44
6.3.4. Zhodnocení	47
7. Inverzní kinematická úloha	48
7.1. Popis nohy robota dle Denavit-Hartenbergovy konvence	48
7.2. Odvození řešení inverzní kinematické úlohy	49
7.3. Poznámka k výpočtu různých řešení	51
7.4. Zhodnocení	52
8. Závěr	53
8.1. Výsledky	53
8.2. Náměty na pokračování	53
9. Literatura	54
10. Příloha – schémata a osazení	55
11. Příloha – výkresy mechanických dílů	60

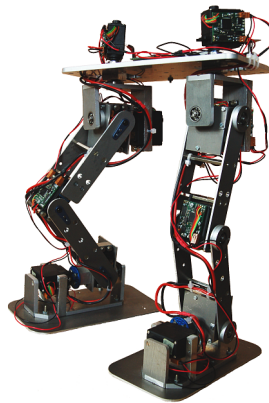
1. Úvod

Hlavním cílem práce bylo vyrobit a řídit skutečného robota, ne jen pouhý matematický model. Východiskem byl nejjednodušší druh dvounohé chůze – statická chůze s pevnou trajektorií. Cíle bylo dosaženo již s regulátorem typu P, ručně tvořenou trajektorií a nedokonalou mechanickou konstrukcí i elektronikou servomotorů. Počáteční chůze a její řízení byly na úrovni běžných hračkařských humanoidních robotů (Robonova, Graupner aj.).

Snažil jsem se jít v návrhu dál a dát k dispozici dobrý hmatatelný základ pro další matematický (softwarový) vývoj. Robot je navržen jako distribuovaný, rozšiřitelný systém. Nezbytnou páteří celého robota je průmyslová sběrnice. Každému servomotoru přísluší jedna počítačová deska, poskytující poměrně slušný výpočetní výkon pro implementaci rychlého místního řízení. Celek se dorozumívá obousměrně s hlavním řídicím počítačem se zpětnou vazbou uzavřenou přes průmyslovou sběrnici.

Mechanický návrh byl veden snahami o minimální pružnost, jednoduchost kinematického popisu a malou hysterezi kloubů, aby bylo umožněno smysluplné zpětnovazební řízení.

V návrhu regulátorů byl po neblahých zkušenostech položen základ řízení, odolnému vůči přetížení.



1. fotografie robota *Spejbla*

2. Dvounohá chůze

2.1. Definice

Dvounohá chůze je pohybem, při němž vždy alespoň jedno z chodidel *stojí* na podlaze. Rozlišujeme fázi, kdy robot stojí pouze na jedné noze (*single-support phase*) a druhou fázi, kdy robot stojí na obou nohou (*double-support phase*). Otázkou zůstává, co přesně znamená, že chodidlo stojí na podlaze. Stejně asi přijmeme klouzání (šoupání) za platnou součást chůze. Chodidlo by tudíž mělo držet na jednom místě podlahy díky třecím silám, které vyrovnávají reakční síly, vyvolané pohybem robota.

K řešení dvounohé chůze existují dva odlišné přístupy: chůze s chvilkovým (*instantaneous*), nebo dlouhodobým (*non-instantaneous*) stojem na obou nohou. Chůze s chvilkovým stojem na obou nohou je mezním případem chůze, blízkým běhu (tj. pohybu, při němž vždy nejvýše jedno z chodidel stojí na podlaze). Tento přístup vynucuje použití dynamické chůze [↔2.2.2.], neboť přenesení těžiště mezi chodidly nemůže být provedeno v nekonečně krátkém čase. Chůze s chvilkovým stojem na obou nohou je základní myšlenkou mnoha teoretických modelů dvounohých robotů [1]. V současnosti mi ovšem není znám žádný skutečný robot, který by byl schopen bez vnější pomůcky¹ takto chodit.

2.2. Trajektorie chůze

2.2.1. Definice

Upřesněme, co nazýváme v řešení dvounohého robota *trajektorií*. Během pohybu robota je každá jeho součást (tuhé těleso) urychlována silami podél nějaké cesty v prostoru.

Robota je možno chápat jako sériový manipulátor s 12 otočnými klouby [↔3.]. Jako manipulátor má 12 stupňů volnosti a jako celek ještě dalších 6 stupňů volnosti, může-li se pohybovat volně v prostoru, tj. není-li žádné z jeho těles „uzemněno“. Trajektorie (v smyslu světových souřadnic) všech bodů robota mohou být tudíž popsány vektorovou funkcí rozměru 18. Přesto však volíme jednodušší popis trajektorie, popisující pouze vzájemnou polohu těles robota mezi sebou. Taková trajektorie neobsahuje žádnou informaci o celkové poloze robota vůči okolnímu světu, má tedy jen 12 stupňů volnosti. Nazýváme ji dále jen *trajektorií* a definujeme polohami (úhly natočení) v $n = 12$ kloubech, $q(t) : \langle 0, t_F \rangle \rightarrow \mathcal{Q}, (q_1, q_2, \dots, q_n) \in \mathcal{Q}$, kde t je čas, t_F je doba chůze a q_k je úhel natočení k . kloubu, \mathcal{Q} je prostor všech přípustných konfigurací kloubů. Jedná se o trajektorii v prostoru *kloubových souřadnic*.

Je nutné mít na paměti, že stručnější trajektorie o 12 stupních volnosti nepopisuje pohyb robota úplně, neboť výsledná poloha robota záleží kromě této *neúplné* trajektorie ještě na zvnějšku působících silách. Např. robot jdoucí po podlaze je posouván dopředu třecí reakční silou podlahy, kdežto robot, který při konání téže trajektorie v kloubových souřadnicích visí ve vzduchu nad podlahou, se vpřed nijak neposune.

Tato trajektorie může být dostatečným popisem v případě, že všechny vnější síly budou známé (tření podlahy, směr tíže) a homogenní (podlaha musí být rovinná, ale může být nakloněná). Z těchto předpokladů vychází veškeré další úvahy, ve většině případů ještě

¹ Robot Rabbit je schopen chůze, ovšem vyžaduje otočnou tyč nebo zábradlí, aby udržel rovnováhu směrem do stran (stranný pohyb nebyl záměrně tvůrčí skupinou řešen).

omezené na vodorovnou podlahu. Vlivy veškerých skutečných odlišností od těchto předpokladů, jako jsou křivost či sklon podlahy, jsou chápány jako vnější rušivé síly, působící na soustavu.

2.2.2. Statická a dynamická chůze

Rozdělení trajektorií na statické a dynamické je obecnou otázkou z oboru mobilní robotiky [2]. Trajektorie může být rozložena na cestu $q(s) : \langle 0, 1 \rangle \rightarrow \mathcal{Q}$, spojitou funkci, tvořící parametrickou křivku v \mathcal{Q} parametrizovanou s , a průběh času (*time-scaling*) $s(t) : \langle 0, t_F \rangle \rightarrow \langle 0, 1 \rangle$, $\dot{s} \geq 0$, tak, že trajektorie $q(t) = q(s(t))$.

Pro danou cestu $q(s)$ může existovat mnoho přípustných průběhů času. Průběh času nazýváme přípustným tehdy, když existuje funkce silových momentů $u(t) = (u_1, u_2, \dots, u_n)$, splňující omezení akčních členů $|u_k(t)| \leq u_{k_{max}}$ taková, že po působení těchto momentů v kloubech vykoná robot trajektorii $q(t)$. Předpokládáme-li určitou cestu $q(s)$, stavový prostor systému je zmenšen na skalární polohu s a rychlost \dot{s} . Omezení akčních členů spolu s dynamikou systému kladou omezení na zrychlení, $L(s, \dot{s}) \leq \ddot{s} \leq U(s, \dot{s})$. Křivka v prostoru (s, \dot{s}) splňující tato omezení určuje potom přípustný průběh času $s(t)$ pro danou cestu.

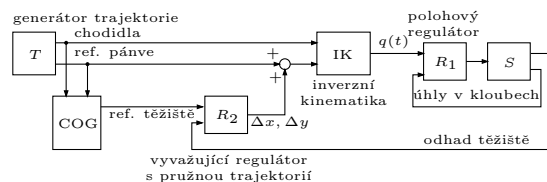
Cesta je *statická*, pokud křivka $\dot{s} = 0$, $s \in \langle 0, 1 \rangle$ určuje přípustný průběh času. To znamená, že robot může stát v libovolné z poloh $q(s)$, aniž by spadnul, aniž by se pohyboval. Robot se může pohybovat po statické cestě libovolně pomalu. V každém bodě $q(s)$ splňuje robot podmínku *statické stability*. Trajektorie, probíhající podél této cesty průběhem $s(t)$ tak, že \dot{s} se blíží nule, je nazývána *statickou trajektorií*. Pokud průběh času s $\dot{s} = 0$ není přípustný, ale jiný, kde $\dot{s} > 0$, hovoříme o *dynamické trajektorii*.

Pokud statická či dynamická trajektorie spolu s vnějšími silami tvoří chůzi [↔2.1.], jedná se potom o *statickou*, případně *dynamickou chůzi*. V této práci jsem se zaměřil zejména na chůzi statickou, která bude též předpokládána v dalším textu, nebude-li řečeno jinak.

2.2.3. Pevná a pružná trajektorie

U robota rozlišují řízení s *pevnou* a *pružnou trajektorií*. Při řízení s pevnou trajektorií je dána $q(t)$ a systém je řízen tak, aby tuto trajektorii co nejlépe sledoval. Úloha je tak převedena na úlohu řízení dynamického systému.

Při řízení s pružnou trajektorií je $q(t)$ určena pouze částečně, ponechávajíc volnost pro průběžnou úpravu trajektorie při chůzi samotné. Cílem je lépe vyrovnávat rušivé vnější síly. Tato pružnost zavádí ve své podstatě do systému další zpětnou vazbu. Jsou-li již motory uzavřeny v smyčce polohové zpětné vazby, tvoří kolem ní pružná trajektorie vnější zpětno-vazební smyčku, dohromady kaskádní regulátor.



2. příklad regulátoru s pružnou trajektorií

S pružnou trajektorií je možno lépe potlačit silné poruchy, jako např. proměnný sklon podlahy nebo vliv nákladu na polohu těžiště. Příklad jednoduchého regulátoru pro pružnou trajektorii je na obr. 2. Trajektorie je dána kartézskými souřadnicemi chodidel a pánve.

Skutečná poloha těžiště je odhadována z údajů čidel (měřený sklon, tlak v chodidlech, odběr motorů) a porovnávána s referenční polohou těžiště. Rozdíl je veden přes regulátor zpět, aby vychýlil pánev o $\Delta x, \Delta y$ oproti referenční poloze. Pánev tak vyrovnává rušivé vlivy, aby udržela těžiště v stabilní poloze. Aby mohl regulátor pracovat v jiných než kloubových souřadnicích, je třeba v reálném čase počítat inverzní kinematickou úlohu [↔7.].

Nejsložitějším způsobem úpravy trajektorie je její úplné pozměnění, např. kladení chodidel do jiných míst tak, aby bylo zabráněno pádu. Nejsou mi známi žádní skuteční dvounozí roboti pracující na tomto principu. Takové chování bylo ovšem již s úspěchem použito u jiných krácejících robotů, např. u čtyřnohého robota firmy Boston Dynamics [3].

Ve své práci jsem používal pouze chůzi s pevnou trajektorií.

2.2.4. Tvorba trajektorie

Trajektorie statické chůze sestává ze dvou částí: přenesení průmětu těžiště z půdorysu jednoho chodidla nad druhé a přenesení volného chodidla na jiné místo podlahy. Poté následuje zrcadlově shodná posloupnost, v níž jsou role nohou vyměněny.

Trajektorie může být vytvořena ručně „animací“. Několik bodů $q(s_k), k = 1 \dots m$ je ručně sejmuto (navzorkováno) např. tak, že robot je „zmrazen“ zpětnou vazbou v bodě $q(s_k) = konst.$ a pokud se tato poloha ukáže být staticky stabilní [↔2.2.2.], může se tento bod stát součástí vznikající trajektorie. Po sejmutí všech bodů následuje *interpolace v kloubových souřadnicích*. Takto vytvořená trajektorie může selhat ze dvou důvodů:

- příliš řídké vzorkování – body vzniklé interpolací mezi body sejmutými nemusejí nutně splňovat podmínky statické stability;
- příliš vysoká rychlost – přípustnost statické trajektorie je zaručena pouze pro nekonečně pomalou chůzi, při vyšší rychlosti ji může zhatit setrvačnost robota.

Tvorba trajektorie může být usnadněna řešením inverzní kinematické úlohy [↔7.]. Při ruční animaci je tak možno nastavovat polohu v jednotlivých bodech tak, aby byla dodržena některá daná omezení, např. aby roviny chodidel byly rovnoběžné s rovinou pánve, aby byla chodidla navzájem rovnoběžná atd. Místo nastavování kloubů je pak polohována pánev, případně pánev a chodidlo v kartézských souřadnicích (a případně Eulerových úhlech natočení).

3. Mechanická konstrukce

Mechanická konstrukce robota je inspirována součástmi a pohybovými možnostmi lidských nohou. Robot je tvořen dvěma nohama, z nichž každá má 6 stupňů volnosti. Žádné další stupně volnosti v robotu obsaženy nejsou. Nahoře končí tělo robota pánví. Inspirací pro tento způsob řešení dvounohého robota, bez možnosti vyvažovat rovnováhu nakláněním vrchní části těla, nám byl zejména robot YABiRo [4]. Trojrozměrný model robota je na obr. 3 (oproti konečné verzi jsou na obrázku kratší kosti a scházejí sněžnice).

Každá noha je tvořena velkým stabilním chodidlem, umožňujícím statickou chůzi, při níž je těžiště udržováno nad plochou opěrného chodidla, případně mezi chodidly. Odlišný přístup nalézáme u robotů s bodovými chodidly, zejména u vícenohých (čtyřnohých jako např. [5] nebo velmi rozšířených šestinohých), případně u výlučně dynamicky kráčejících dvounohých robotů [1]. Přítomnost velkého chodidla ovšem nijak nevyklučuje dynamickou chůzi.

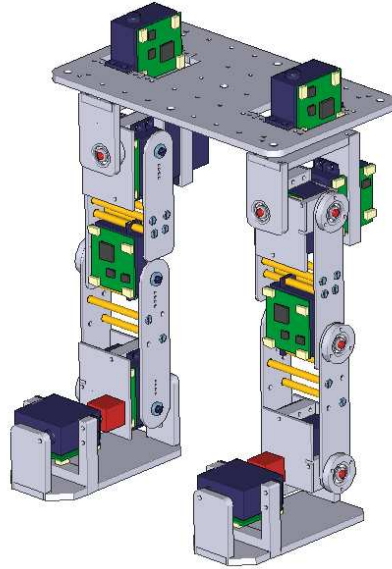
Chodidlo je připojeno k holeni dvouosým kotníkovým kloubem, holeň k stehnu jednoosým kolenním kloubem a stehno k pánvi, spojující obě nohy, tříosým kyčelním kloubem. Víceosé klouby (kotníkový a kyčelní) se skládají z několika jednoosých kloubů, jejichž osy otáčení se protínají v jediném bodě a jsou navzájem kolmé. Cílem použití víceosých kloubů bylo zjednodušit kinematický popis robota, především umožnit kvalitativně jednodušší řešení inverzní kinematické úlohy [↔7.]. Tato výhoda schází v mnoha návrzích modelářských robotů i u robota Pino [6].

Každá z os kloubů je poháněna shodným typem modelářského servomotoru, HSR-5995TG výrobce Hitec, u něhož byla původní elektronika nahrazena naším vlastním návrhem [↔4.3.2.]. Servomotor má být schopen vyvinout moment 2,4 Nm, což bylo vedle malých rozměrů (délky v ose) určujícím činitelem jeho volby. Kryt modelářského servomotoru a jeho rozměry byly východiskem celého mechanického návrhu. Všechny osy jsou poháněny přímým napojením na hřídel motoru, táhla byla vyloučena pro omezení pružnosti v dynamice soustavy. Tento přístup je běžný u modelářských robotů a u robota Pino [6] a liší se od robota YABiRo [4], jenž používá u všech servomotorů táhel.

3.1. Konstrukční řešení

Všechny osy pohybu vyjma svislé osy kyčelního kloubu jsou uloženy na jedné straně v ložisku servomotoru, na druhé straně v samostatném kuličkovém ložisku. Ve víceosých kloubech otáčí jeden servomotor pouzdrem, v němž je zabudován druhý servomotor, otáčející navazujícím ramenem v druhé, kolmé ose. Protilehlé ložisko schází u svislé osy kyčelního kloubu. Ta je uložena v dvou kuličkových ložiscích umístěných na pánvi těsně nad sebou, sloužících zároveň jako axiální uložení.

Návrh mechanických dílů byl veden snahou co největší podíl součástí nechat vypálit dvourozměrným CNC laserem. Většinu součástí tvoří výpalky z 2 mm, 5 mm a 10 mm duralového (AlCu4Mg) plechu. Dále jsou použity distanční sloupky s vnitřním závitem M3, ložiska 625, hřídelky koncem o čtvercovém průřezu, kvádříky kotníkové osy, metrické šrouby M3, M2,5, M2 a M1,6.



3. trojrozměrný CAD model robota

3.2. Popis sestavení

Výkresy mechanických součástí robota jsou v příloze [↔11.].

Obě nohy jsou navzájem zrcadlově symetrické. Klouby kyčle i kotníku obsahují pouzdro na servomotor, tvořené deskami (C8, C9 v případě kotníku a K5, K6 v případě kyčle), dvěma špalíčky (C6) a stěnou C7. Do děr ve špalíčcích je vyříznut závit M3 a přišroubována k nim plastová příruba servomotoru. Desky kryjí větší stěny servomotoru, z jedné strany přiléhá deska těsně, na druhé je ponechána mezera 7 mm na desku elektroniky. K deskám je z jedné strany přišroubována základna hřídelky C5, z druhé rozhraní připojovací páku navazujícího servomotoru (kvádrík C10 u kotníku, K4 u kyčle). Obojí je k deskám přišroubováno šrouby M3 se zápustnou hlavou (hlavy z vnitřní strany desek). Ve stěně C7 je vyfrézována slepá díra průměru 10 mm a hloubky 2,5 mm, v jejímž středu je vypilován čtvercový otvor 3×3 mm. Do díry je zapuštěn válcový výběžek servomotoru, umístěný výrobcem v ose otáčení. Do závitů M3 výběžku je zašroubována hřídelka N6, procházející (a proti protáčení zajištěná) čtvercovým otvorem v stěně C7. Desky jsou ke špalíčkům C6 a stěně C7 přišroubovány šrouby M2,5 (v C6, C7 vyříznuty závity). Hřídelka, spojená s pouzdry servomotoru, je řešena spojením výpalku C5 a hřídelek C11 (chodidlo), K8 (kyčel) závitěm M5 (snaha vyloučit soustruh).

Základnu chodidla tvoří deska C1a s děrami průměru 3 mm pro připevnění sněžnice a 2,5 mm pro přišroubování svislých dílů C2, C3, C4 šrouby M2,5 se zápustnou hlavou, pro něž je v dílech C2, C3, C4 vyříznut vnitřní závit. Servomotor číslo 1 je přišroubován přírubou k C3 opatřeným závitěm M3 a vnitřním závitěm válcového výběžku k C2. Páka servomotoru 1 je k pouzdru servomotoru 2 připojena kvádríkem C10. Původní jednostranná páka výrobce Hitec zde byla nahrazena neznačkovou pákou ve tvaru kotoučku, přichycenou k C10 čtyřmi šrouby. Díl C4 je ložiskovým pouzdrům pro ložisko 625. Servomotor 2 otáčí navazující holení.

Holeň a stehno jsou tvořeny díly N2a, N4a a N3a, N5a (byla vyrobena i o čtvrtinu kratší podoba dílů N2, N4, N3, N5, které nakonec nebyly použity). N4a, N5a na vnitřní straně nohy jsou pevně spojeny s pákami servomotorů šrouby M2 do vnitřního závitě v páce (nutno

zvětšit jednu z děr ve výpalcích) a M1,6 s matickou do díry bez závitů. K N2a, N3a jsou připevněna šrouby M2 se zápustnou hlavou pouzdra N1 s ložiskem 625 (vnitřní závit v N1). N2a oproti N4a a N3a oproti N5a jsou rozepřeny distančními sloupky délky 49mm s vnitřním závitem M3. Podobnými distančními sloupky délky 30,5 mm a distančním sloupkem bez závitů délky 16mm je k stehnu přišroubován kolenní servomotor číslo 3. Ve válcovém výběžku servomotoru 3 je zašroubována hřídelka N7, procházející čtvercovým otvorem v N3a.

V kyčelním pouzdru se nachází servomotor číslo 4, otáčející stehnem. Pouzdem otáčí servomotor číslo 5. Jeho původní jednostranná páka byla k desce K5 připevněna přes rozhraní K4 třemi šrouby M3. Páka i s K4 byla následně nahrazena pákou ve tvaru kotoučku a připevněna ke K5 čtyřmi šrouby M2,5 se zápustnou hlavou, zašroubovanými do závitů v kotoučku páky. Servomotor 5 je přírubou přišroubován k rámu K3 (vnitřní závit M3), který je spolu s ložiskovým pouzdem K2 na ložisko 625 přišroubován k horní desce K1 šrouby M2,5 (vnitřní závit v K2, K3). V původním návrhu byla K1 připevněna přes K7 přímo k páce servomotoru číslo 6 (svislá osa). Toto napojení bylo velmi nedokonalé a tvořilo značnou hysterzi kyčelního kloubu, která činila chůzi velmi nespolehlivou a animaci obtížnou ([↔6.2.3.]). Proto byl kyčelní kloub upraven. Svislá osa byla uložena do dvou kuličkových ložisek 625 těsně nad sebou. Jejich pouzdem je díl K9, připevněný čtyřmi šrouby M2,5 k desce K8. Ložisky prochází hřídel K10, dole přišroubovaná k desce K1 a nahoře napojená prostřednictvím dílu K11 k hřídeli servomotoru. Na ní je nasazen váleček, vytvořený zbrúšením páky do kulata.

V pánvi P1 byly přírubami přišroubovány (vnitřní závit M3 v P1) servomotory číslo 6 obou nohou s přírubou na horní straně pánve, ve verzi s opravenou kyčlí jsou k P1 přišroubovány desky K8 a servomotory jsou proti protáčení v svislé ose drženy kolíky, opět zašroubovanými v P1 (nezakresleno). Pánev umožňuje volit rozteč os kyčelních kloubů v rozsahu 100 až 140 mm, při sestavování robota byla zvolena rozteč 120 mm.

Sněžnice byly při sestavování připevněny šrouby M3 se zápustnou hlavou a matkami k chodidlům, širším okrajem blíže ke středu robota a zadní straně.

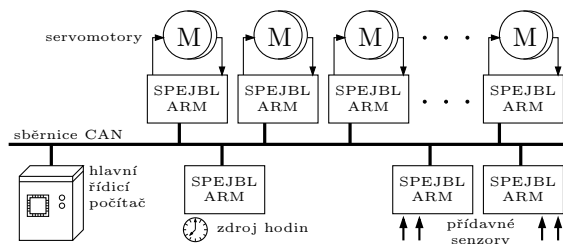
4. Elektronika

Každý z 12 servomotorů je vybaven destičkou „SPEJBL-ARM“, vyrobenou v rámci této práce, osazenou 32-bitovým jednočipovým mikro počítačem s jádrem ARM a vyvedenou průmyslovou sběrnici CAN (Controller Area Network) [7]. Deska SPEJBL-ARM je v každém servomotoru propojena s výkonovou a měřicí elektronikou. Sběrnice CAN je připojena k hlavnímu řídicímu počítači, z něhož je robot ovládán.

Schémata zapojení jsou v příloze [↔10.].

4.1. Počítačová síť

Desky SPEJBL-ARM příslušející k servomotorům tvoří 12 uzlů sítě CAN (obr. 4), představující na síti jak uzly akčních členů, tak čidel polohy servomotorů. Do systému mohou být přidány další senzory, např. zrychloměry (sklonoměry) nebo čidla tlaku v chodidlech. Dalším uzlem sítě je hlavní řídicí počítač, přijímající měřené údaje o poloze z desek servomotorů (případně i jiné veličiny z ostatních senzorů) a rozesílající řídicí povely zpět do desek servomotorů. Jako řídicí počítač byl použit počítač typu IBM PC, který je možné nahradit libovolným vestavěným systémem (procesor RISC, FPGA), připojitelným ke sběrnici CAN.



4. uzly počítačové sítě

Přínos průmyslové sběrnice v návrhu robota:

- měřené analogové veličiny nejsou vedeny na velké vzdálenosti v blízkosti silných pulzních pohonných proudů;
- jednoduchá celková kabeláž, sestávající z pouhých 4 vodičů (2 napájecí a 2 pro sběrnici CAN), nepřekáží příliš v pohybu kloubů;
- snadná rozšiřitelnost systému o další prvky;
- volnost ve volbě řídicího počítače.

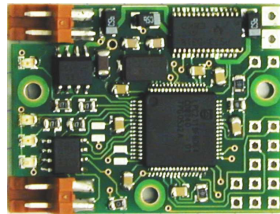
Sběrnice CAN byla zvolena pro dostupnost a nízkou cenu součástek a zařízení, širokou hardwarovou podporu v jednočipových počítačích a v ovladači OCERA LinCAN [8], použitím v řídicím počítači. Další deska SPEJBL-ARM byla zapojena do sítě jako generátor synchronního časování pro všechny ostatní uzly, *time-trigger* (hodiny).

Vzhledem k vysokému příkonu je robot napájen z nepohyblivého zdroje (7 V, 10 A) kabelem. Teoreticky by bylo možné robota napájet i z baterií, nejspíše z lithiových akumulátorů. Vzhledem k potížím s výdrží a změnami parametrů těchto akumulátorů stárnutím a též nízké nosnosti již nyní poněkud přetíženého robota jsem se v řešení nesnažil o samostatně se pohybujícího robota. Tomu odpovídá i přímé připojení sběrnice CAN k řídicímu počítači kabelem.

4.2. Deska SPEJBL-ARM

Destička SPEJBL-ARM (obr. 5) byla navržena přednostně pro řízení servomotoru robota Spejbla. Je ovšem vhodná i na jiné použití. Mezi požadavky návrhu byla potřeba malých rozměrů, určených velikostí modelářského servomotoru, sběrnice CAN a volnosti při výběru jiných typů motorků či snímačů (pro případ použití synchronních motorů nebo inkrementálních čidel v budoucnu). Vlastnosti destičky:

- rozměry $37 \times 28 \times 6,5$ mm (bez konektorů $37 \times 28 \times 3,5$ mm);
- mikročítač Philips LPC2119 s jádrem ARM7TDMI;
- napájení $5,5 \text{ V} \dots 18 \text{ V}$, $< 30 \text{ mA}$ (bez servomotoru; pro motor je horní mez cca 7 V !);
- sběrnice CAN 2.0, 5 V , až 1 Mb.s^{-1} ;
- vstupy/výstupy v $3,3 \text{ V}$ úrovních, 5 V tolerantní; vyvedeno $6 \times$ PWM, $4 \times$ A/D, $7 \times$ vstup záchytného čítače, $2 \times$ UART (multiplexováno na 10 vývodech).



5. deska SPEJBL-ARM v měřítku 1:1

Základem desky je jednočipový mikročítač s jádrem ARM, jmenovitě Philips LPC2119, běžící na frekvenci do 60 MHz . Jádro ARM7TDMI provádí celočíselné výpočty v 32-bitové aritmetice. To dává desce dostatečný výpočetní výkon pro implementaci rychlých místních smyček řízení motorů. Volba tak silného jednočipového mikročítače se může zdát přehnanou, ale v době návrhu byla obchodní cena 8-bitového jednočipového mikročítače vybaveného řadičem CAN rovna ceně LPC2119.

Zavádění a ladění programu je možno provádět po vyvedené asynchronní sériové lince s použitím vnitřního ISP² zavaděče LPC2119. Rozhraní JTAG není na destičce vyvedeno. V robotu Spejblovi je z praktických důvodů používáno zavádění programu přes CAN (vlastní zavaděč je ovšem naprogramován do paměti FLASH přes sériové ISP). K vynucení spuštění ISP zavaděče je vyveden signál ISPSEL#.

Kromě vstupního napětí je na destičce stabilizováno 5 V pro budič CAN PCA82C250 a $3,3 \text{ V}$ a $1,8 \text{ V}$ pro LPC2119. Stabilizaci $3,3 \text{ V}$ a $1,8 \text{ V}$ provádí spolu s vytvářením signálu RESET# kombinovaný obvod TPS73HD318. Obě napájení jsou větvena mezi analogovou a číslicovou část.

Vstupní napájecí napětí i sběrnice CAN jsou vyvedeny na dvoupólové konektory PS25/2G. Konektory byly zvoleny pro své malé rozměry při poměrné bytelnosti a proudové zatížitelnosti. Zbylé vývody jsou v robotu Spejblovi bez konektorů, použité pouze jako pájecí plošky. Jsou ale osaditelné konektory s roztečí $2,54 \text{ mm}$.

Všechny součástky jsou osazeny povrchovou montáží na jedné straně dvouvrstvé desky plošných spojů v 6. konstrukční třídě ($\geq 7 \text{ mil}$ mezera, $\geq 9 \text{ mil}$ tloušťka spoje). Pro konektory K1, K2 jsou po stranách desky vyfrézovány obdélníkové výřezy, konektory jsou připájeny kolíky rovnoběžně s deskou, povrchově.

² in-system programming

V desce jsou vyvrtány tři otvory pro uchycení šroubem M2, zároveň je jedna strana ponechána hladká, bez součástek, aby ji bylo možné přilepit. V robotu jsou destičky přišroubovány na stěnách servomotorů, šrouby se zdrsňenou válcovou hlavou jsou přilepeny kyanakrylátovým lepidlem.

4.3. Elektronika servomotoru

V servomotoru je použit komutátorový („stejnoseměrný“) motorek s převodovkou, poloha výstupní hřídele je snímána potenciometrem. Zdroj stejnosměrného napájecího napětí pro motoru je zdrojem napájení celého robota. Výše tohoto napětí, zvolená 7 V, je určena podle údajů výrobce servomotoru (uvádí rozsah 4,8...7,4 V). Motor je řízen napájením přes H-můstek, spínaný logickými signály pulsně-šířkové modulace (PWM).

4.3.1. Původní elektronika

Původní řídicí elektronika (jednočipový mikropočítač) modelářského servomotoru Hitec HSR5995-TG nespĺňovala naše požadavky, zejména pro

- nedostatek vhodného rozhraní (CAN či jiné sběrnice, umožňující připojení více uzlů);
- chybějící snímání polohy;
- nemožnost implementace vlastních rozmanitých regulátorů.

V první verzi jsem se rozhodl nahradit původní řídicí mikropočítač deskou SPEJBL-ARM a původní výkonovou elektroniku ponechat. Schéma zapojení původní elektroniky servomotoru a jejího připojení k desce SPEJBL-ARM je v příloze.

H-můstek, spínající napájení motoru, je tvořen dvěma integrovanými dvojicemi MOSFET Vishay Si9986DY či novějším typem Si4532ADY. Každá z dvojic je buzena jedním logickým signálem. Vzhledem k vysoké kapacitě hradel použitých MOSFET (1 nF) je měkké buzení bipolárními tranzistory s rezistory v kolektoru nedostačující a vybití hradla trvá příliš dlouhou dobu. Po tu dobu působí oba tranzistory větve proti sobě do zkratu. V původním servomotoru s frekvencí PWM 1 kHz byl patrně občasný zkrat snesitelný, při zvýšení na 20 kHz již působil značný klidový odběr a zahřívání tranzistorů.

Celek je ponechán na původní desce plošného spoje uvnitř servomotoru, na místo původního jednočipového mikropočítače ATMEGA8L jsou připájeny vodiče se signály PWM. Elektronika servomotoru je propojena s řídicí destičkou SPEJBL-ARM sedmi krátkými vodiči.

S touto elektronikou byla provedena první úspěšná chůze, použitý regulátor byl typu P, [↔6.2.]. Pro značné potíže s přehříváním tranzistorů MOSFET byla pak tato sebezničující elektronika nahrazena následující verzí.

4.3.2. Vlastní elektronika

Zapojení desky SPEJBL-ARM s původní elektronikou servomotoru stále nevyhovovalo potřebám, proto jsme ji nahradili vlastním návrhem³. Důvody byly:

- chybný návrh, nedbající proudových omezení (výkonové tranzistory často shořely při větší zátěži na hřídeli motoru);
- scházející měření proudu, užitečné pro odhadování zátěžového momentu a potažmo vnějších sil (tíže);
- pomalé buzení hradel MOSFET;

³ patrně i sám výrobce Hitec zjistil vady výrobku, proto jej poměrně brzy po uvedení stáhl z prodeje a nahradil jej novým typem

- nemožnost úplně odpojit motor (při zkratu zůstává elektromechanicky zabrzděn a ztěžuje ruční animaci).

Zásadní změnou bylo doplnění obvodu o měření proudu, které zabezpečuje sériový odpor $0,11\ \Omega$, tvořený dvěma rezistory R10, R11 a rozdílový zesilovač U5. Měřený proud je převáděn na napětí úměrně velikosti rezistoru R9. Tento signál obsahuje velké množství vyšších harmonických PWM, proto je filtrován RC článkem R9, C1 [↔6.3.1.1.]. Zapojení měří absolutní hodnotu proudu, tekoucího ze zdroje do H-můstku. Omezení proudu bylo uskutečněno softwarově, takže v elektronické části je pouze jeho měření.

Budicí tranzistory U3, U4 zůstaly podobné, byla zvolena novější verze Si4532ADY. Buzení jejich hradel bylo posíleno zvlášť pro tento účel vyráběnými budiči U1, U2. Projistotu byl ještě zaveden nepatrný ochranný interval ($1\ \mu s$) mezi přepnutím horního a dolního tranzistoru, sloužící k bezpečnému vybití hradla. Proto je buzen každý MOSFET zvlášť logickým signálem, k buzení jsou tak celkem zapotřebí 4 logické signály. To zároveň umožňuje úplné odpojení motoru od zdroje a jeho protáčení bez elektrického brzdění zkratem.

Logické vstupy budičů jsou ochráněny hradlem v diodové logice (D1, D2, R5, R6, R7, R8), které zamezuje smrtící kombinaci obou tranzistorů dvojice sepnutých a ponechává pouze 3 zbývající (horní sepnut, dolní sepnut a odpojeno). Při odpojených vstupech je implicitně dolní tranzistor sepnut v obou větvích.

Po neblahých zkušenostech s hořením elektroniky byl rovněž na desku zahrnut teploměr s polovodičovým termistorem U6.

Deska byla vytvořena v tloušťce laminátu a s obrysovým frézováním tak, aby padla dovnitř servomotoru na místo desky původní.

4.3.3. Přetrvávající chyby v návrhu

Popsaný návrh je bohužel rovněž špatný z hlediska přesného měření proudu. Chyby v měření proudu jsou:

- Unipolární měření – je měřena pouze absolutní hodnota proudu. Při řízení nelze jednoznačně určit znaménko proudu.
- Měření proudu H-můstkem, nikoli motorem – je měřen proud, který jde z napájecího zdroje do motoru, nikoli proud vinutím motoru. Z hlediska proudové ochrany se jedná o řešení používané, ale z hlediska přesného zpětnovazebního řízení, případně odhadování zátěžového momentu, je takto měřený údaj chybný. V případě, že jsou póly motoru spojeny nakrátko tranzistory H-můstku, může motorem protékat brzdný proud, který ovšem nebude změřen.

Správnou opravou první chyby je měřit proud obou polarit s hladkou návazností v oblasti kolem nulového proudu. Řešením druhé chyby je měřit tento proud na takových místech H-můstku, aby odpovídal skutečnému proudu vinutím. To je velice podstatné zejména pro případ, že by mělo být měřeného proudu využito pro odhad zátěžových momentů, následně vnějších sil.

5. Software

5.1. Provoz na síti a časování

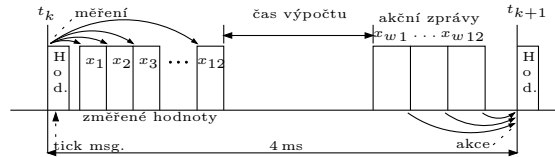
Desky SPEJBL-ARM běží bez operačního systému, využívají přímo přerušeni. Desky řídicí servomotory počítají řízení v rychlých místních zpětnovazebních smyčkách, jmenovitě v smyčce proudového regulátoru [↔6.3.1.]. Dále měří polohu a proud a vyměňují si veličiny s hlavním řídicím počítačem přes sběrnici CAN. Měření proudu probíhá s vzorkovací frekvencí 120 kHz, snímání polohy a proudová zpětná vazba jsou vzorkovány s frekvencí 20 kHz.

Deska hodin (time-trigger) působí na sběrnici jako *master*, poslouchána všemi ostatními uzly sítě včetně hlavního řídicího počítače. Její úlohu by mohl plnit kterýkoli jiný uzel sítě s dostatečně přesným časováním. Vzhledem k tomu, že není vyžadována od řídicího počítače, nemusí jeho hardware a operační systém splňovat tak přísné požadavky na odezvu v reálném čase. Deska hodin vysílá zprávy („ticky“) na frekvenci 250 Hz, sloužící systému jako synchronní vzorkovací frekvence, umožňující všem uzlům měřit vstupní veličiny a nastavovat akční veličiny v téměř okamžiku.

Vzorkování vnitřních smyček v deskách SPEJBL-ARM je asynchronní vůči hodinám sítě CAN. Vzniká tak chyba časování (jitter). Vzhledem k značnému poměru vzorkovací frekvence vnitřní smyčky oproti hodinové frekvenci sítě ([↔6.3.2.1.]) pokládám takto vzniklou chybu za zanedbatelnou z hlediska funkce řídicího systému. Synchronizaci by bylo možné provést přizpůsobením vnitřních hodin desky SPEJBL-ARM hodinovým zprávám na sběrnici CAN. Zároveň by bylo možné vzájemně posunout náběžné hrany PWM různých motorů do přesně určených rozestupů, což by omezilo nárazový pulzní odběr z napájecího zdroje. Řešení otázky synchronizace jsem se ale v práci nevěnoval.

V každém z implementovaných diskretních regulátorů se odráží konečná doba výpočtu, případně ještě přenosu po sběrnici CAN. Akční veličina nemůže být uplatněna dříve, než po uplynutí této doby. V návrhu číslicového řízení se používají dva přístupy k zapracování této doby do návrhu – buď uplatnit akční zásah co nejdříve a zavést zpoždění, rovné necelé vzorkovací periodě, nebo počkat do okamžiku dalšího vzorku a uplatnit akční veličinu se zpožděním přesně 1 krok. Ve všech regulátorech robota jsem použil druhý přístup, čímž získávám zpoždění z^{-1} a výhodu přesného časování, v němž se neprojeví přerušeni ani provoz na síti.

Hlavní řídicí počítač běží pod operačním systémem Linux 2.6, v provedených pokusech bez jakýchkoli rozšíření pro práci v reálném čase (RT Linux nebo fully preemptive kernel). U programu byla nastavena statická priorita reálného času (nejvyšší možná). Na odezvu počítače v reálném čase jsou kladeny poměrně mírné požadavky jednak díky tomu, že hlavní řídicí počítač není zdrojem celkové synchronizace (time-triggerem), a také tomu, že bylo zvoleno počkat mezi vzorkováním měřených veličin a uplatněním akčních zásahů v řídicí smyčce celou jednu periodu provozu na síti CAN (4 ms). Bezchybná činnost bez ztráty vzorků vyžaduje, aby byly všechny akční zásahy, které je třeba vyslat přes síť, spočteny a odeslány mezi posledním po síti poslaným měřením k . vzorku a příchodem $(k + 1)$. hodinové zprávy, tedy v časovém úseku dlouhém 68 % periody provozu na síti CAN, tj. přibližně 2,72 ms.



6. provoz na sběrnici CAN během jedné periody

Provoz na síti CAN je znázorněn na obr. 6. Deska hodin (time-trigger) posílá periodicky hodinové zprávy („tiky“). Ty obsahují 8-bitové číslo pořadí, aby mohli ostatní účastníci (hlavní řídicí počítač) zjistit případnou ztrátu vzorků. V okamžiku příchodu hodinové zprávy jsou vzorkována současně všechna čidla (potenciometry servomotorů), případně jsou zaznamenány jejich v daném okamžiku nejčerstvější hodnoty. Sejmuté hodnoty jsou posílány přes síť CAN hlavnímu řídicímu počítači. Každý uzel, jehož se měření týká, použije svůj vlastní CAN ID (identifikátor). Následně hlavní řídicí počítač určí hodnoty akčních veličin pro příští vzorkovací okamžik. Může tak učinit na základě přijatých měření, nebo bez ohledu na ně, podle toho, zda je použita regulace typu MIMO v hlavním počítači, nebo SISO v každé desce servomotoru zvlášť [↔6.1.]. Akční veličiny jsou rozeslány po síti deskám servomotorů. Pro úsporu přenosové kapacity sítě je posíláno méně akčních zpráv než je motorů a jedna zpráva obsahuje hodnoty pro více motorů najednou (konkrétně 4 16-bitové hodnoty akční veličiny v jedné zprávě).

Sběrnice CAN je nastavena na nejvyšší možnou přenosovou rychlost, $1 \text{ Mb}\cdot\text{s}^{-1}$. Perioda provozu ($1/f_{CAN} = 4 \text{ ms}$) byla zvolena z datové propustnosti při rozumné celkové zátěži sítě. Horní odhad celkové zátěže sítě počítá s úplným stuffingem⁴ (natažením délky oblastí rámce, jichž se stuffing týká, na $6/5$ délky, viz [7]). V jedné periodě je třeba přenést 1 hodinovou zprávu (1 byte), 12 zpráv od servomotorů (po 2 bytech), 3 akční zprávy (po 8 bytech). Pro délku periody 4 ms vychází zátěž sběrnice CAN do 33,2%, pokud jsou zprávy vysílané servomotory rozšířeny na 6 byte (s přidavkem hodnot napětí a měřeného proudu), zátěž sběrnice bude do 44,6%.

5.2. Programování SPEJBL-ARM

Mikro počítač LPC2119, osazený na desce SPEJBL-ARM, obsahuje 16 KB statické paměti RAM a 128 KB paměti FLASH. Pracuje na taktovací frekvenci do 60 MHz.

V současnosti je paměť RAM schopna s velkou rezervou pojmout data i samotný kód programu, proto jsou vždy po spuštění nahrávány programy do RAM desek SPEJBL-ARM z hlavního řídicího počítače. Nahrávání probíhá zavaděčem vlastní výroby přes sběrnici CAN. Vysílací strana zavaděče používá knihovnu `libbfd` pro čtení „binárních“ (zkompilovaných) souborů formátu ELF, z něhož kromě umístění a obsahu paměťových bloků programu vyčte i adresu, z níž má být program spuštěn. Tento zavaděč je schopen nahrávat pouze do RAM, v principu ale nic nebrání rozšíření o zápis do paměti FLASH či začlenění ladicího kódu (např. tzv. „stub“ debuggeru GDB) pro možnost ladění po sběrnici CAN.

Každá deska SPEJBL-ARM je při prvotním zápisu přijímací strany zavaděče do paměti FLASH opatřena jedinečným číslem (identifikátorem), který je zároveň identifikátorem na sběrnici CAN v průběhu zavádění programu zavaděčem. Tento identifikátor je dostupný uživatelskému programu, takže program jej může používat jako vlastní CAN identifikátor a zároveň může zjistit, na které konkrétní desce sám běží. Řídicí počítač v okamžiku zavádění zvolí, do kterých desek zavést jaký program (například do jedné destičky zdroj hodin,

⁴ proměnnou výplní fyzické vrstvy

do desek u motorů shodný regulační algoritmus) a program spuštěný v desce může ještě přizpůsobit svou činnost (parametry) na základě identifikátoru.

Software pro desky SPEJBL-ARM byl vyvíjen zcela v jazyce ANSI C, překládán kompilátorem GCC 3.4.3 za podpory sestavovacího prostředí OMK (OCERA Make System⁵).

5.3. Ovládací program

Ovládací program hlavního řídicího počítače slouží k řízení a tvorbě trajektorií chůze robota. Program umožňuje:

- přehrát trajektorii z textového souboru s užitím lineární interpolace v kloubových souřadnicích;
- zaznamenat trajektorii nebo její část v reálném čase, dovoluje animovat robota jako loutku, zatímco motory jsou odpojeny a pouze poloha je snímána;
- polohovat jednotlivé klouby myši, stiskem kláves nebo číselným zadáním polohy;
- zmrazit libovolnou podmnožinu kloubů, dovoluje polohovat ručně zbylé klouby;
- změřit aktuální polohu robota a sejmut tak jeden bod vytvářené trajektorie.

Jsou-li regulátory uspořádány jako SISO [↔6.1.], ovládací program plní pouhou funkci přehrávače (sekvenceru) trajektorií. Je-li použito MIMO uspořádání, regulátor je součástí ovládacího programu.

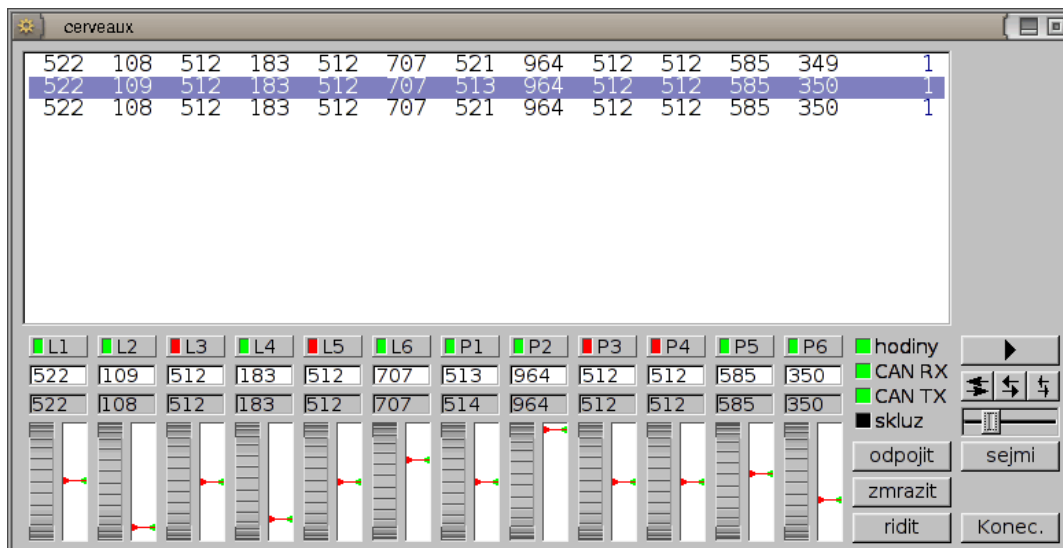
Základem programu je stavový automat, čekající na příchozí zprávy sběrnice CAN. Každá nová perioda vzorkování začíná příchodem hodinové zprávy. Po jejím příchodu čeká program, až přijdou zprávy s měřením od všech servomotorů (obecně čidel). Po příjmu poslední zprávy je měření úplné a regulátor může vypočítat hodnoty akčních zásahů pro příští periodu, které po dokončení výpočtu odešle na sběrnici CAN. Tím vnější působení končí a program čeká na příští hodinovou zprávu. Pokud přijde hodinová zpráva dříve, než zprávy měření, nastala chyba servomotoru (čidla). Pokud se číslo přijaté hodinové zprávy neshoduje s číslem předchozí, zvýšeným o 1 modulo 2^8 , program ohlásí ztrátu vzorků (skluz). To může nastat, když je vykonání programu v hlavním řídicím počítači něčím příliš zbrzděno.

Ovládací program byl vytvořen v základních dvou verzích. První byl dávkový program nazvaný Cerveau, který buď nastavoval data poloh z textového souboru jako žádanou trajektorii nebo naopak snímal polohu a zaznamenával její průběh (animaci) do textového souboru. Interpolace byla řešena pomocným programem mimo Cerveau. Druhou verzí je interaktivní program CerveauX, obsahující grafické uživatelské rozhraní. Oproti programu Cerveau mu schází pouze možnost přímého záznamu animované trajektorie v reálném čase.

5.3.1. Popis programu CerveauX

Uživatelské rozhraní programu je vidět na obr. 7. V dolní části je 12 ukazatelů a ovladačů jednotlivých servomotorů. Rysky ukazují žádanou (x_w) a měřenou x polohu (natočení) hřídelí. Otočným páskem je možno polohu měnit, stejně tak je možno zadat hodnotu do textového políčka. Spínače s kontrolkou umožňují motor odpojit od řízení, kontrolka ukazuje, zda motor odpovídá zprávami po síti. Kontrolky stavu sítě v pravé části ukazují, zda jsou v časovém limitu přijímány hodinové zprávy, zda funguje vlastní vysílání a příjem a zda nedošlo ke skluzu (ztrátě vzorků). Tlačítka umožňují všechny motory hromadně odpojit, zapojit (řídit) a nebo zmrazit, což znamená, že je nejprve jejich požadovaná poloha nahrazena hodnotou měřenou $x_w = x$ a pak je zapnuto řízení. Zbývající tlačítko způsobí sejmutí aktuální polohy a její vepsání do textového okna nahoře.

⁵ <http://rtime.felk.cvut.cz/omk/>



7. uživatelské rozhraní ovládacího programu CerveauX

Textové okno programu v sobě chová polohy (body) tvořené či přehrávané trajektorie. Každý řádek textu odpovídá jednomu vzorku (bodu), každý z prvních 12 sloupců čísel odpovídá polohám jednotlivého motoru. Číslo v 13. sloupci je čas [s], po který trvá interpolace z hodnoty v příslušném řádku do hodnoty v řádku následujícím. Čísla, popisující polohy motorů, jsou $x_w \in \mathbb{Z}, 0 \leq x_w < 1024$ a jejich jednotkou je LSB⁶.

Samotné textové okno pracuje jako plnohodnotný textový editor se všemi běžnými funkcemi (přesuny, kopírování, používání schránky). Je v něm možné psát libovolný text, ovšem program předpokládá, že jsou v řádcích textu obsaženy buď platné vzorky trajektorie nebo komentáře. Za platnou řádku trajektorie je považován takový řádek, který se skládá z 12 celých a 1 reálného čísla, obklopených případně libovolnými bílými znaky. Program provádí při editaci i kopírování neustálý syntaktický rozbor změněných částí textu a na základě toho vyhodnocuje jeho obsah. Zároveň provádí zvýrazňování textu podle syntaxe a rovněž upozorňuje barevně na hodnoty mimo rozsah.

Je-li řádka syntakticky správná, může být označena (stiskem myši) jako aktivní a v tu chvíli je její obsah propojen s ovládacími prvky motorů. Její obsah se přenáší do ovladačů motorů a zpětně i jejich ovládání se promítne do změn hodnot v textu. Text stále zůstává textem, je možno jej plně měnit i kopírovat. V případě sejmutí poloh je buď změněn obsah aktivního řádku, nebo je ze sejmutých hodnot vytvořen řádek nový, pokud žádný řádek aktivním nebyl. Textové okno programu tedy obsahuje pracovní trajektorii. Tu je možno v podobě textu přenášet z a do jiných programů pomocí schránky (X Window System – X Selection).

Při přehrávání trajektorie dochází u posledního řádku ke kruhovému návratu na řádek první. Přehrávání je možné kdykoli pozastavit a v pozastavené podobě ručně plynule posouvat v rámci interpolačního intervalu.

Stavový automat síťového provozu s výpočtem interpolace a regulátoru běží ve vlákně s vysokou prioritou. Uživatelské rozhraní programu běží s obyčejnou prioritou a je k prioritnímu vláknu vázáno asynchronně a poměrně volně, aby neomezilo jeho odezvu v reálném čase. Textový editor používá syntaktický rozbor tvořený rovněž stavovým automatem. Grafické uživatelské rozhraní je postaveno na knihovně FLTK 1.1.7⁷. Připojení k síti CAN

⁶ least significant bit, velikost rozlišení A/D převodníku

⁷ <http://fltk.org/software.php?VERSION=1.1.7>

obstarává knihovna VCA (Virtual CAN API⁸ [8]), použitým ovladačem v jádře systému Linux 2.6 byl LinCAN verze 0.3.

⁸ <http://www.ocera.org/download/distributions/partialrel/lincan-0.3-rc1.html>

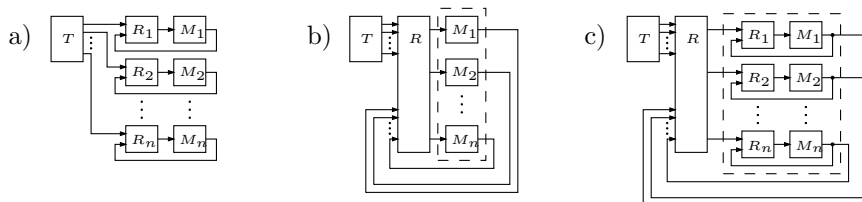
6. Zpětnovazební řízení

Z hlediska řízení dynamického systému je robot mechanickým systémem sestávajícím z tuhých těles, pružnost hřídelí motorů zanedbáváme. Podle Lagrangeovského modelování obsahuje stavový vektor systému jako celku 12 poloh (úhlů) a 12 (úhlových) rychlostí, systém je 24. řádu. Síly působící na soustavu těles jsou jednak vnitřní: 12 třecích sil v kloubech, a vnější: 12 momentů, působených motory, reakční a třecí síla podlahy, tíže a další poruchové síly. Výstupem systému je 12 úhlů natočení kloubů $x_1 \dots x_{12}$, cílem je přiblížit tyto úhly referenční trajektorii, $x_{wk} = q_k$.

6.1. Rozdíly SISO a MIMO řízení

K řízení dvounohých robotů existují dva přístupy. Jednodušší z nich spočívá v použití *několika SISO*⁹ regulátorů, jeden regulátor na každý motor. Každý motor je tak řízen *nezávisle* svým vlastním regulátorem (obr. 8a). Nevýhodou tohoto přístupu je, že vzájemné dynamické působení těles robota není zohledněno, regulátory nemohou „spolupracovat“. Výhodou je jednoduchost.

Při SISO řízení může být každý regulátor nastaven odlišně, podle svého umístění (kloubu, kde působí). Rovněž mohou být v průběhu chůze měněny jeho parametry podle toho, v které části chůze (kroku) se robot právě nalézá. V nejjednodušším případě je ale možno nastavit všechny regulátory zcela shodně, tak, aby byly dostatečně robustní pro regulaci dynamiky v libovolném kloubu a v libovolném okamžiku chůze, ovšem za cenu nižší kvality regulace. Typickou volbou regulátoru pro SISO řízení je PID regulátor, případně kaskáda PID regulátorů.



T – generátor trajektorie (vysílá $x_{wk} = q_k$), $R, R_1 \dots R_n$ – regulátory,
 $M_1 \dots M_n$ – motory

8. a) SISO, b) MIMO, c) kaskádní MIMO-SISO regulátor

Jinou možností je použít místo několika regulátorů *jeden MIMO*¹⁰ regulátor (obr. 8b). Akční veličinou je vektor, jehož složky působí na jednotlivé motory. Měřená poloha natočení ve všech kloubech je vstupem regulátoru a může být doplněna údaji z ostatních čidel pro zpřesnění odhadu polohy uvnitř MIMO regulátoru (pokud jsou taková čidla, např. sklonoměry nebo čidla tlaku v chodidlech, přítomna). Akční veličina jakožto výstup MIMO regulátoru by teoreticky mohla přímo nastavovat činitel plnění napěťové PWM motorů. V praxi je ovšem nutné, aby motor byl uzavřen alespoň ve smyčce regulátoru proudu, chránící jej a výkonovou elektroniku před zničením nadproudem. Výsledkem je MIMO-SISO kaskádní regulátor (obr. 8c).

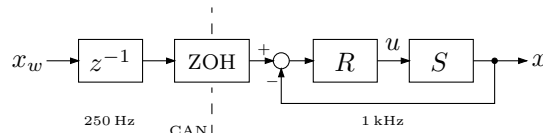
⁹ single-input, single-output, tj. jeden vstup, jeden výstup

¹⁰ multiple-input, multiple-output, tj. mnoho vstupů, mnoho výstupů

Výhodou MIMO řízení je, že může odrážet vzájemné dynamické působení uvnitř řízeného systému a že umožňuje spojení měřených dat z potenciometrů v servomotech a z ostatních čidel (fúzi dat). MIMO regulátorem může být např. spojení pozorovatele a zpětné vazby navržené jako LQG nebo ručním umístěním pólů. V teorii dobře rozpracovaným regulátorem pro úlohu řízení dvouhého robota je bytostně nelineární regulátor, postavený na základě pozorovatele s rychlou dynamikou (high-gain observer) a stavové zpětné vazby s přesnou linearizací [9] [10].

6.2. P regulátor

V první verzi robota byl použit prostý proporcionalní (P) regulátor v místní SISO smyčce polohové zpětné vazby. Měřenou a referenční veličinou byla poloha (úhel) v kloubu x , akční veličinou byl přímo činitel plnění napěťové PWM u , což odpovídá první verzi elektronického zapojení [↔4.3.1.]. Software regulátoru běžel uvnitř desky SPEJBL-ARM, smyčka byla uzavřena místně a síť CAN sloužila pouze pro přenos referenční veličiny (obr. 9).



9. blokový diagram P regulátoru

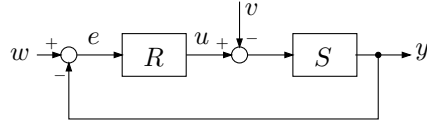
6.2.1. Rozbor

SISO regulátor řídí servomotor bez ohledu na ostatní servomechanismy v systému. Lineární model servomotoru, ze kterého jsem vycházel, počítá se zanedbatelnou pružností hřídele, zanedbatelnou elektrickou časovou konstantou motoru (což je v případě použitého motoru opodstatněné, viz [↔6.3.1.]) a setrvačnou zátěží hřídele. Výsledkem je model 2. řádu, obsahující časovou konstantu T_m , danou součinitelem lineární složky tření v kloubu a momentem setrvačnosti těles spojených s hřídelí, a astatismus, integrující rychlost na polohu:

$$S(s) = \frac{K}{s(1 + T_m s)}.$$

Časová konstanta T_m , úměrná setrvačné hmotě v daném kloubu, se značně mění podle toho, o který kloub se jedná a dále rovněž podle aktuální polohy robota – pracovního bodu, v němž uvažujeme linearizovaný model. Velmi zásadní změna setrvačnosti v některých kloubech (zejm. kotníkových) nastává, když dochází ke změně ze stoje na obou nohou na stoj na jedné noze. Cílem bylo nastavit regulátor tak, aby uspěl ve všech kloubech a všech polohách při stejném nastavení proporcionalní konstanty r .

Charakteristický polynom uzavřené smyčky s proporcionalním regulátorem $R(s) = r$ je $\Phi(s) = T_m s^2 + s + rK$. Nutnou a postačující podmínkou stability systému 2. řádu je, aby byly všechny koeficienty charakteristického polynomu téhož znaménka, což je splněno pro všechna $r > 0$. Pro různé setrvačné hmoty se bude tedy (v lineárním přiblížení) měnit doba ustálení, stabilita bude zachována ve všech případech.



10. porucha vstupující do uzavřené smyčky

Z hlediska regulační odchylky e v ustáleném stavu sledujeme přenos $F_{w \rightarrow e}$ z reference x_w na odchylku a přenos $F_{v \rightarrow e}$ z poruchy na vstupu v na odchylku (obr. 10). Po dosažení dostáváme:

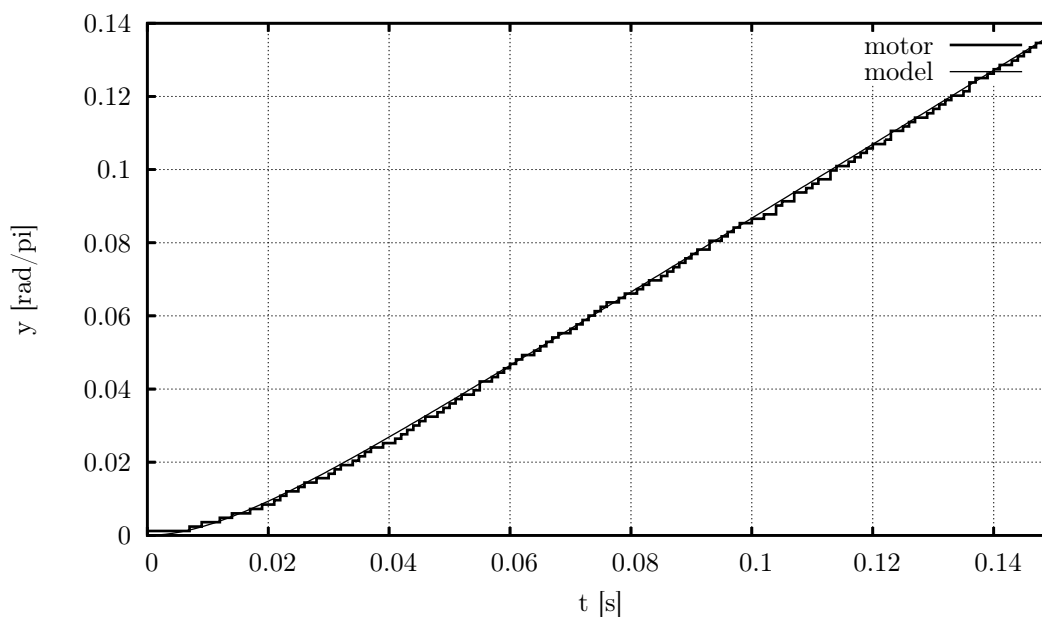
$$F_{w \rightarrow e} = \frac{1}{1 + RS} = \frac{s(1 + T_m s)}{T_m s^2 + s + rK}, \quad F_{v \rightarrow e} = \frac{S}{1 + RS} = \frac{K}{T_m s^2 + s + rK}.$$

Regulační odchylky pro referenční a poruchovou veličinu n . řádu odpovídají přenosům stejnosměrné složky $\lim_{s \rightarrow 0} F(s)/s^{n-1}$. Přenos z konstantní reference (1. řádu) na odchylku je $\lim_{s \rightarrow 0} F_{w \rightarrow e} = 0$, z reference typu rampa (2. řádu) je to $\lim_{s \rightarrow 0} F_{w \rightarrow e}/s = 1/rK$. Přenos konstantní poruchy na odchylku je $\lim_{s \rightarrow 0} F_{v \rightarrow e} = 1/r$. P regulátor zajistí regulaci konstantní reference s nulovou regulační odchylkou, což potřebám statické chůze vyhovuje. Při konstantní poruše na vstupu, kterou v případě krácejícího robota tvoří zejména statické zátěžové momenty, působené tíží, ovšem vznikne nenulová regulační odchylka, což je i u statické chůze nežádoucí. Pro její odstranění je třeba regulátor rozšířit o integrační složku.

Jak již bylo řečeno, velikost setrvačných hmot na hřídeli servomotoru se velice liší pro různé klouby a okamžiky chůze. Parametry servomotoru byly identifikovány pouze pro jedinou (a to poměrně malou) setrvačnou zátěž, a sice pro samotný servomotor (se značnou vlastní setrvačností, tvořenou zejména převodovkou) s připojeným ramenem N5a. Cílem bylo zejména zjistit dolní odhad časové konstanty pro správnou volbu vzorkovací frekvence [11]. Časová konstanta byla určena z odezvy na jednotkový skok (obr. 11) a to nelineární regresi 1. diference výstupního signálu funkcí $y(t) = b_1(1 - e^{-b_2 t})$. Úloha nelineární regrese s kritériem nejmenších čtverců byla vyřešena Marquardt-Levenbergovou metodou, implementovanou v programu Gnuplot¹¹. Důvodem použití nelineární regrese oproti lineární regresi ARX modelu byl fakt, že lineární regrese dávala vinou kvantizačního šumu zcela zcestné výsledky. Vypočtená časová konstanta je $T_m = 1/b_2 = 14,3 \cdot 10^{-3}$ s. Elektromechanické statické zesílení motoru bylo určeno ze sklonu několika ustálených odezev na jednotkový skok pro různé vstupní činitele plnění. Výsledný statický přenos pro činitel plnění napěťové PWM $u \in \langle -1, 1 \rangle$ jako vstupní veličinu a úhel x jako výstup vyšel roven $K = 2,30 \pi$ rad.

¹¹ <http://www.gnuplot.info/>

Prechodova charakteristika samotneho motoru (PWM 44%)



11. přechodová charakteristika motoru

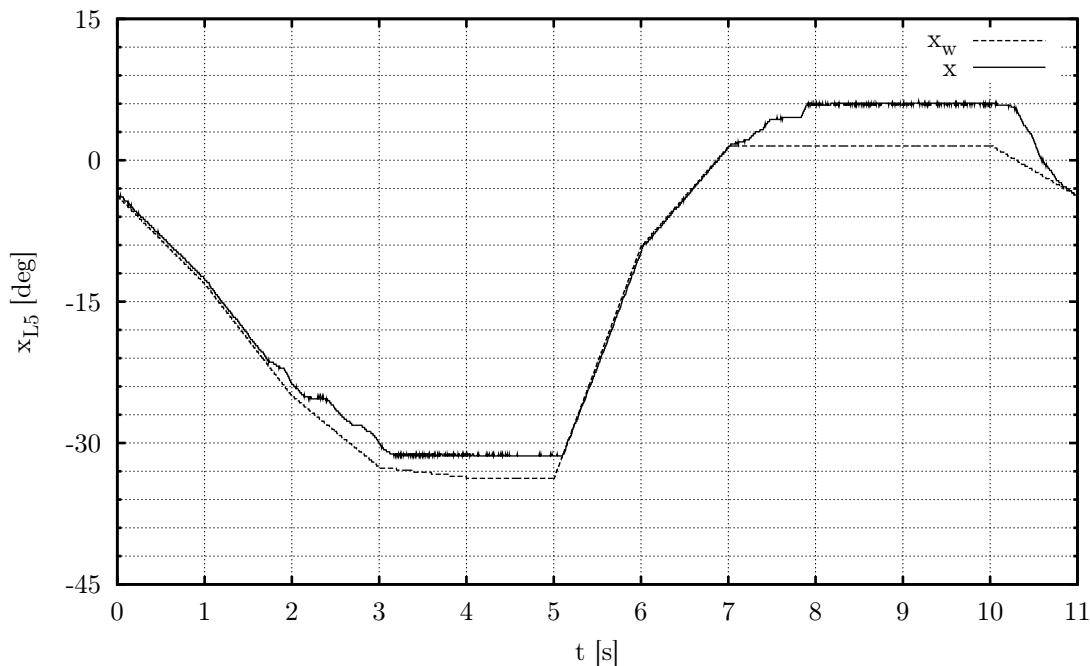
Ze změřených odezev na různě veliké vstupní skoky u bylo zjištěno, že pásmo necitlivosti motoru, dané Coulombovským třením, je téměř neznamenné. Z odezev bylo rovněž patrné, že šum měření polohy potenciometrem je zanedbatelný vzhledem k nerovnoměrnosti dráhy potenciometru a nepřevyšuje ± 1 LSB. Kvantizace měřené polohy činí 832 LSB na π rad.

6.2.2. Realizace

Proporční konstanta regulátoru byla nastavena empiricky na $r = 15 \pi^{-1} \text{rad}^{-1}$. Regulátor byl realizován diskrétně, vzorkovací frekvence byla zvolena 1 kHz ($4 \times$ vzorkovací frekvence na síti CAN), tzn. že vzorkovací perioda je cca. $14 \times$ kratší než T_m . Frekvence PWM byla zvolena z akustických důvodů 20 kHz, vzorkování měřené polohy bylo provedeno koherentně s PWM. Vzorkovací frekvence regulátoru nebyla synchronizována s vzorkovací frekvencí sítě CAN, neboť změny referenčních průběhů jsou dostatečně pomalé a časová chyba v rozmezí $1/4$ periody není závažná.

6.2.3. Výsledky

P regulátor splnil svou úlohu ve všech kloubech a po celou dobu chůze, při extrémně setrvačných polohách (nakročená noha vzhledem k protějšímu kotníku) a při rychlé chůzi docházelo k zdatelným zákmitům. Ukázka referenční a řízené trajektorie pro jeden kloub je na obr. 12. Chůze se zdařila.



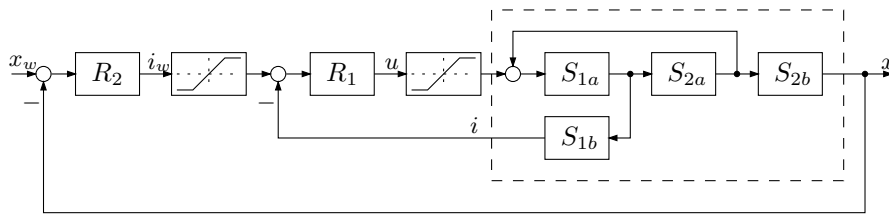
12. řízení trajektorie v laterální ose levého kyčelního kloubu

Dle očekávání byla při gravitačním zatížení zratelná regulační odchylka, ta byla ovšem v první verzi robota zanedbatelná vůči jiné, přičítající se chybě, způsobené mechanickou hysterezi v kotníkovém a zejména kyčelním kloubu. Vinou tohoto jevu sledoval robot jinou než referenční trajektorii a již při animaci bylo nutno ji nastavovat s ohledem na tuto chybu. Bohužel tato systematická chyba podléhala navíc změnám během opotřebení zmíněných kloubů a to činilo úspěch chůze nejistým. Na základě této vady došlo k přepracování mechaniky kloubů [↔3.2.].

Nejzásadnější vadou tohoto prvotního řešení bylo časté proudové přetěžování motoru a s ním spojené výkonové elektroniky. K němu docházelo zejména brzdnými proudy při velkých mechanických zátěžích a když byl motor napájen napětím v okamžicích blízkých mechanickému zkratu. Výsledkem bylo kromě přehřívání vinutí též hoření výkonových tranzistorů, tání přilehlých plastů, samovolné odpájení součástek z desky apod. Pro odstranění této vady bylo rozhodnuto vytvořit vnitřní smyčku regulátoru s proudovým omezením [↔6.3.1.].

6.3. Kaskádní PID-PI regulátor

Jak z neblahých pokusů s první verzí regulace vyplynulo, je včlenění proudového omezení do servomechanismu nutné pro jeho sebezáchovu. Z hlediska zpětnovazebního řízení dynamických systémů je omezení proudu vhodné realizovat kaskádním regulátorem s uměle vloženým omezením („saturací“) akční veličiny – proudu [12], obr. 13. Regulátor R_1 vnitřní smyčky je regulátorem proudu, jeho akční veličinou je napětí (u nás napěťová PWM). Regulátor R_2 vnější smyčky je již polohovým regulátorem a jeho akční veličinou je proud, který je uměle (softwarově) omezen – tím je docíleno, že proud, řízený vnitřní smyčkou, nepřesáhne dané meze, až na případný překmit (jsou-li splněny předpoklady, kladené na řád referencí a poruch).



13. blokový diagram kaskádního regulátoru

6.3.1. Proudový PI regulátor

Úprava elektroniky pro kaskádní řízení spočívala v doplnění obvodu o měření proudu [↔4.3.2.], regulátor je jinak celý tvořen programem v desce SPEJBL-ARM. Regulátor vnitřní proudové smyčky R_2 přijímá žádanou hodnotu proudu i_w , která je pro jistotu oříznuta (omezena), $|i_w| < i_{max}$. Velikost maximálního přípustného proudu i_{max} je dána softwarem v desce SPEJBL-ARM. Akční veličinou proudového regulátoru je napětí (PWM) u . Toto napětí vstupuje do vinutí, v němž vytváří proud, který je filtrován RC článkem (viz dále [↔6.3.1.1.]) a měřen jako i . Proud dále působí v motoru otáčivý moment, zrychlení, rychlost a mění polohu x . Otáčení nebo brzdění motoru, ať už způsobené vzbuzeným proudem nebo vnější mechanickou zátěží, způsobuje zpětnou indukci napětí do vinutí a to se přičítá v podobě zpětné vazby k akčnímu napětí u . Pro důsledné vyjádření dynamiky mezi působeným napětím u a vzbuzeným proudem i bychom tuto zpětnou vazbu měli vzít v úvahu. Ale vzhledem k tomu, že mechanické děje jsou řádově pomalejší (T_m) než odezva, již budeme vyžadovat od proudové smyčky, můžeme chápat obě části systému motoru odděleně s tím, že zpětně indukovaná napětí se projeví jako (pomalu se měnící) porucha na vstupu vinutí u .

Napětí u není možno z podstaty PWM měnit častěji, než s každou periodou PWM, tj. na vzorkovací frekvenci rovné frekvenci PWM; tu vzhledem k režii přepínání tranzistorů H-můstku není rozumné příliš zvyšovat, proto jsem se přidržel akusticky dané hodnoty $f_{PWM} = 20 \text{ kHz}$. Na téže vzorkovací frekvenci nechť tedy pracuje proudový regulátor. Dynamika řízené soustavy je tvořena dynamikou vinutí a měřicího RC filtru. Z měření střídavých vlastností motoru vyplynulo, že na frekvencích do 10^5 Hz se vinutí chová jako čistě reálná zátěž (indukčnost nezelezné kotvy je minimální, samorezonance motoru byla naměřena okolo $3 \cdot 10^5 \text{ Hz}$). Z hlediska regulátoru je vinutí pouze reálnou vodivostí K_1 . Vybuzený proud je pak filtrován RC členem, sloužícím k potlačení vysokých harmonických pulzního PWM buzení, které by znatelně zkreslily údaje o měřeném proudu. Zjednodušený přenos řízeného proudového systému je tedy $S_1(s) = K_1/(1 + T_{RC}s)$.

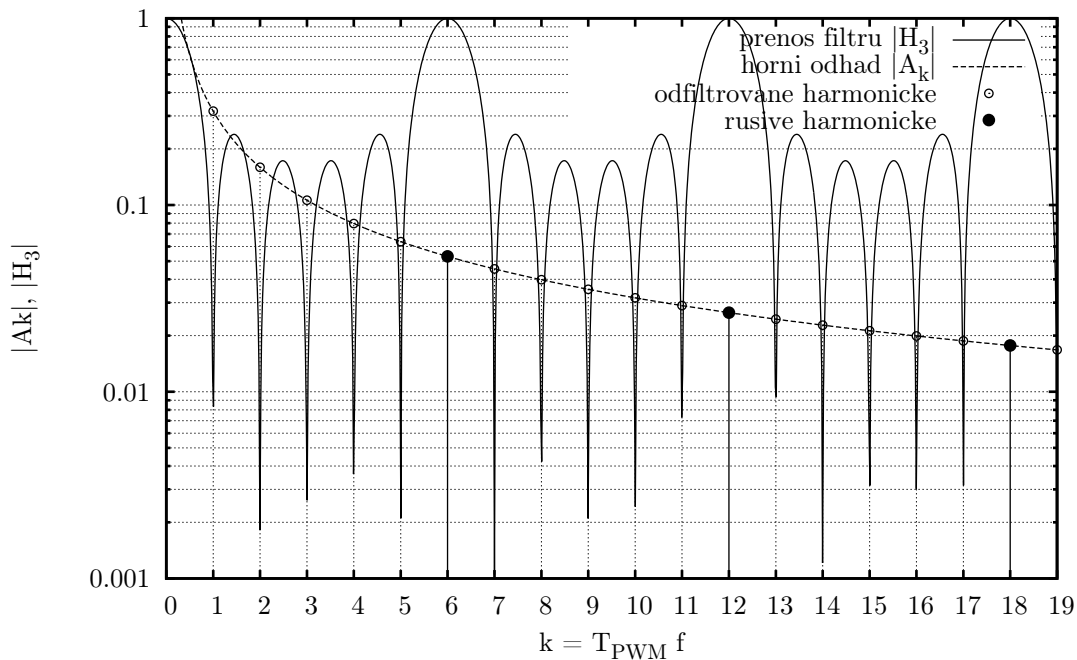
Řízený systém S_1 je stabilní, bez astatismu. Abychom dosáhli nulové regulační odchylky v případě konstantní reference i poruchy na vstupu, musí regulátor obsahovat alespoň jeden astatismus. Zvolil jsem tedy regulátor typu PI, v spojitě podobě $R_1 = (r_0s + r_{-1})/s$. To dává přenosy z referenční veličiny a z poruchy na vstupu:

$$F_{w \rightarrow e} = \frac{s(1 + T_{RC})}{T_{RC}s^2 + (1 + K_1r_0)s + K_1r_{-1}}, \quad F_{v \rightarrow e} = \frac{K_1s}{T_{RC}s^2 + (1 + K_1r_0)s + K_1r_{-1}},$$

tedy $\lim_{s \rightarrow 0} F_{w \rightarrow e} = 0$ a $\lim_{s \rightarrow 0} F_{v \rightarrow e} = 0$, jak bylo požadováno.

6.3.1.1. Filtrace PWM a převzorkování

Diskrétní regulátor je se spojitým řízeným systémem propojen zadržní na straně akční veličiny a vzorkovačem na straně veličiny měřené. Roli zadržce v tomto případě hraje modulátor PWM. PWM je nedokonalou náhradou zadržce 0. řádu (ZOH). Signál PWM s konstantním činitelem plnění (drženou hodnotou) se skládá ze stejnosměrné složky, rovné požadované držené hodnotě, a vyšších harmonických o frekvencích $kf_{PWM}, k \in \mathbb{N}$. Pokud se činitel plnění mění, může se PWM od ZOH lišit nepatrně i na frekvencích odlišných, než jsou kf_{PWM} – dochází k intermodulaci [13]. Pro náš účel je toto zkreslení ovšem zanedbatelné, většina rozdílového signálu zůstává na vyšších harmonických frekvencích.



14. horní odhad amplitudy vyšších harmonických PWM v měřeném signálu

Měřený proud obsahuje vyšší harmonické PWM. Při vzorkování měřeného proudu dochází k zrcadlení ve spektru (aliasing). Vyšší harmonické složky se zobrazují do dolního pásma a jejich amplitudy se přičítají k hodnotě skutečné stejnosměrné složky, již chceme měřit. Zrcadlením působenou chybou lze omezit lepší filtrací před vzorkováním (anti-aliasing) nebo zvýšením vzorkovací frekvence. Zlepšit potlačení vyšších frekvenčních složek před vzorkováním lze snížením zlomové frekvence filtru nebo zvýšením řádu filtru. Zvyšovat řád filtru jsem nechtěl, aby zůstala zachována jednoduchost obyčejného RC filtru 1. řádu. Příliš nízká zlomová frekvence filtru ovšem zbytečně zúží šířku pásma regulační smyčky. Řešením je tedy zvýšit vzorkovací frekvenci a zbytek filtrace provést již digitálně.

Digitální filtr pracující s navzorkovaným signálem může zcela potlačit některé vyšší harmonické, pokud na jejich frekvencích budou ležet nuly filtru. I při převzorkování (zvýšení vzorkovací frekvence f_s na několikanásobek f_{PWM}) se ale některé harmonické ozrcadlí na nulovou frekvenci a způsobí tak chybu stejnosměrného měření. Na obr. 14 je zobrazen vliv spektra vyšších harmonických ve výsledném signálu pro převzorkování $M = 6, f_s = Mf_{PWM}$. Nuly filtru leží na všech harmonických PWM, kromě $kMf_{PWM}, k \in \mathbb{N}$ (tedy 0., 6., 12., ... harmonické), jejichž fáze se přičítají k stejnosměrné složce.

Odhadněme, jakou největší chybu mohou zrcadla vyšších harmonických způsobit pro dané převzorkování $M = f_s/f_{PWM}$ a zlomovou frekvenci RC filtru $f_{RC} = 1/2\pi RC$. PWM souměrná jako sudá funkce má kosinový rozklad s koeficienty (amplitudami k . harmonické)

$$A_k = \frac{1}{\pi k} \sin \pi k u, \quad |A_k| \leq \frac{1}{\pi k},$$

kde $u \in (0, 1)$ je činitel plnění PWM. Zrcadla se ovšem sčítají fázorově, takže celková chyba (vztažená k rozsahu stejnosměrné složky, $|u_{max}| = 1$) je

$$\Delta = 2 \sum_{k=1}^N A_k \leq 2 \sum_{k=1}^N |A_k|,$$

což příliš nepřekvapí, vzhledem k tomu, že řada $\sum_{k=1}^{\infty} 1/k$ ani nekonverguje (násobení 2 je kvůli oboustrannému spektru). Bohužel, analytické vyjádření skutečné chyby se mi vyjma konkrétních případů ($u = 1/2$) nepodařilo. Dále jsou amplitudy harmonických potlačeny RC filtrem s přenosem

$$H(j\omega) = \frac{1}{1 + j\omega RC}, \quad |H_k| = |H(j2\pi k f_{PWM})| = \frac{1}{\sqrt{1 + \left(\frac{k f_{PWM}}{f_{RC}}\right)^2}},$$

$$\frac{k f_{PWM}}{f_{RC}} \gg 1 : |H_k| \approx \frac{f_{RC}}{k f_{PWM}} \geq |H_k|.$$

Při převzorkování M počítáme pouze harmonické Mk . Horní odhad chyby můžeme učinit z absolutních hodnot fázorů (amplitud), řada je již po RC filtraci konvergentní:

$$\begin{aligned} \Delta &= 2 \sum_{k=1}^{\infty} H_{(Mk)} A_{(Mk)} \leq 2 \sum_k |H_{(Mk)}| |A_{(Mk)}| \leq 2 \sum_k \frac{f_{RC}}{f_{PWM} Mk} \frac{1}{\pi Mk} = \\ &= \frac{2 f_{RC}}{\pi f_{PWM} M^2} \sum_k \frac{1}{k^2} = \frac{\pi f_{RC}}{3 f_{PWM} M^2}. \end{aligned}$$

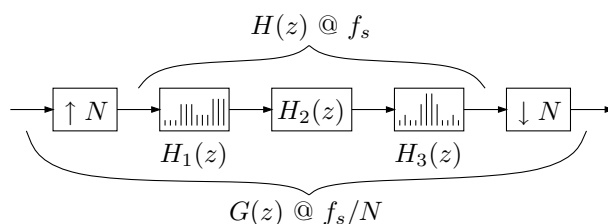
S pomocí numerických výpočtů jsem zkusmo našel o řád nižší horní odhad chyby

$$\Delta \leq \frac{f_{RC}}{8 f_{PWM} M^2},$$

pro nějž bohužel neznám analytické odvození. Odhad se velice těsně blíží skutečnosti pro $f_{RC} < f_{PWM}$ a $M > 1$. V proudovém regulátoru robota bylo použito převzorkování $M = 6$, dané mezní rychlostí vzorkování vestavěného A/D převodníku, a RC filtr $R = 15 \text{ k}\Omega$, $C = 10 \text{ nF}$, $f_{RC} \doteq 1061 \text{ Hz}$. To dává střízlivější (analyticky podložený) odhad chyby $\delta \leq 0,15 \%$ ($\sim -56 \text{ dB}$), realističtější empirický odhad $\delta \leq 0,018 \%$ ($\sim -75 \text{ dB}$).

6.3.1.2. Přenos decimovaného systému

Na straně zádrže (PWM \sim ZOH) vstupuje do řízené soustavy signál na vzorkovací frekvenci f_{PWM} , na straně vzorkovače (A/D) je vzorkován na frekvenci $f_s = M f_{PWM}$. Jedná se o systém se změnou vzorkovací frekvence, signál vzorkovaný f_s je decimován na frekvenci zpětnovazební smyčky f_{PWM} . Obdobná situace nastává potom i ve vnější polohové smyčce, kde je vzorkovací frekvence opět decimována z f_{PWM} na $f_{CAN} = 250 \text{ Hz}$ [\leftarrow 6.3.2.2.]. Odvodíme, jaký je přenos systému s následnou decimací.

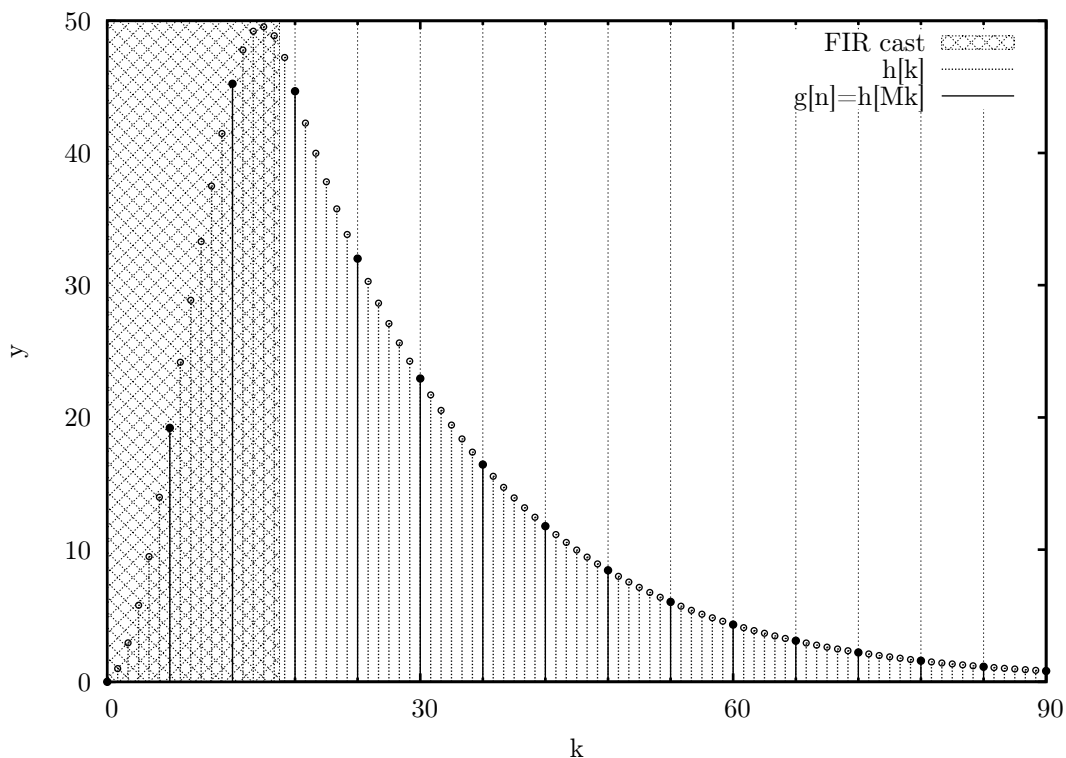


15. blokový diagram systému se změnou vzorkovací frekvence

Uvažujme diskrétní systém s přenosem $H_2(z)$, pracující na vyšší vzorkovací frekvenci f_s (obr. 15). Jeho vstup je buzen vzorky, přicházejícími na nižší vzorkovací frekvenci f_s/M přes blok $\uparrow M$, kde jsou prokládány nulami a interpolátor (zádrž) s přenosem $H_1(z)$. Předpokládáme, že zádrž lze popsat jako LTI¹² systém, což platí pro v řídicí technice běžně používané zádrže 0. a 1. řádu a obecně pro většinu interpolačních filtrů. Na výstupu může být signál ještě filtrován filtrem $H_3(z)$ před decimací $\downarrow M$, kde je vzorkovací frekvence opět snížena vynecháváním vzorků na f_s/M .

Úkolem je určit přenos $G(z)$ celku, jakožto systému podvzorkovaného na f_s/M , tzn. převést přenos $H(z) = H_1(z)H_2(z)H_3(z)$ na nižší vzorkovací frekvenci. Je dobré si uvědomit, že na vyšší vzorkovací frekvenci f_s nelze o přenosu celku uvažovat, neboť prokládání $\uparrow M$ a decimace $\downarrow M$ jsou sice operace lineární, ale nikoli časově invariantní, takže celek není na vzorkovací frekvenci f_s LTI systémem. Z hlediska nižší vzorkovací frekvence f_s/M se ovšem působení obou bloků „vyrovná“ a celek se již zvnějšku jako LTI systém chová, lze jej popsat přenosem $G(z)$ a jako takový je též plně popsán svou impulsovou odezvou. Stačí tedy do systému vpustit jednotkový impuls, který je zádrží $H_1(z)$ „vytvarován“ (zádrž 0. řádu z něj například udělá posloupnost M impulsů shodné výšky), výsledek projde systémem $H_2(z)$ a filtrem $H_3(z)$ a z výstupu (impulsové odezvy $h[k]$) je pak vybrán každý M . vzorek, tedy hledaná impulsová odezva $g[k] = h[Mk]$ (obr. 16). Změna vzorkovací frekvence je zcela obdobná diskretizaci spojitého systému, kdy je jeho přenos nejprve vynásoben LTI modelem zádrže (např. ZOH) a poté popsána impulsová odezva celku jako přenos.

¹² lineární, časově invariantní



16. impulsová odezva systému a její decimace

Je vidět, že impulsová odezva celku $g[k]$ vlastně vzniká pouhou decimací impulsové odezvy $h[k]$. Jaký je vztah mezi oběma přenosy, $H(z)$ a $G(z)$? Vliv vynechávání vzorků (decimace) je dobře patrný, rozložíme-li $H(z)$ na částečné zlomky a decimujeme odezvu každého pólu zvlášť:

$$H(z) = H_F(z) + H_P(z) = \sum_{i=0}^m \frac{b_i}{z^i} + \sum_{i=1}^{n-m} \sum_{j=1}^k \frac{r_{i,k}}{(z - q_i)^j}, \quad q_i \neq 0.$$

V rozkladu jsem záměrně rozdělil přenos na součet části s konečnou impulsovou odezvou (FIR část) $H_F(z)$ a zbylou, přísně ryzí část s nekonečnou impulsovou odezvou (ryzí část) $H_P(z)$. Obě části totiž decimace pozmění odlišně. Nekonečné impulsové odezvy $H_P(z)$ mají tvar obecně komplexních exponenciál, v případě násobných pólů násobených polynomem v čase (čísle vzorku k), z nichž se po decimaci stanou opět exponenciály s patřičně umocněným základem (pólem):

$$h_P[k] = \sum_i \sum_j p_{i,j} k^j q_i^k, \quad g_P[k] = h_P[Mk] = \sum_i \sum_j p_{i,j} M^j k^j (q_i^M)^k.$$

Ryzí část decimovaného přenosu $G_P(z)$ tedy obsahuje umocněné póly q_i^M původního přenosu $H_P(z)$ při zachování jejich počtu a násobnosti. Zajímavá skutečnost nastane při decimaci FIR části $H_F(z)$. Ta popisuje děj s konečnou impulsovou odezvou, trvající v čase $k = 0, \dots, m$. Z této konečné odezvy jsou ponechány pouze vzorky na časových násobcích M , zbytek je zcela ignorován:

$$h_F[k] = b_k, \quad g_F[k] = h_F[Mk] = b_{Mk}.$$

Dochází k výraznému *snížení řádu* FIR části přenosu. Tento jev se uplatní významně, neboť používané zadržky $H_1(z)$ a v našem případě i filtr $H_3(z)$ bývají typu FIR a celek $H(z)$ má poměrně dlouhou odezvu FIR části.

6.3.1.2.1. Algoritmus decimace přenosu

Provádět převod přenosu $H(z)$ na decimovaný $G(z)$ přímo podle uvedeného rozkladu na částečné zlomky je mírně nešikovné. Předně je třeba spočítat samotný rozklad na částečné zlomky a dále poměrně obtížně přepočítat koeficienty u vícenásobných pólů. Jinou cestou je provést decimaci ve stavovém popisu systému:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}u_k,$$

$$\mathbf{x}_{k+M} = \mathbf{A}^M \mathbf{x}_k + \mathbf{A}^{M-1} \mathbf{B}u_k.$$

kde \mathbf{x}_k je vektor stavů systému a u_k vstup systému v čase k . Výstup systému je dán lineární kombinací stavu $y_k = \mathbf{C}\mathbf{x}_k$. Matice systému \mathbf{A} se decimací změní na \mathbf{A}^M a vstupní matice \mathbf{B} se změní na $\mathbf{A}^{M-1} \mathbf{B}$. Tato metoda se vyskytuje v literatuře pod názvem *lifting*, případně v obecnější podobě převodu decimovaného systému na systém s více vstupy a výstupy jako *blocking* či *serial to parallel conversion* [14]. Pro decimaci přenosové funkce je třeba převést ji na stavový popis a zpět, například takto (s omezením na přísně ryzí systém):

$$H(z) = \frac{B(z)}{A(z)}, \quad B(z) = \sum_{k=1}^n b_k z^{-k}, \quad A(z) = 1 + \sum_{k=1}^n a_k z^{-k},$$

$$\mathbf{A} = \begin{pmatrix} -a_1 & -a_2 & \dots & -a_n \\ 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 1 & 0 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad \mathbf{C} = (b_1 \quad b_2 \quad \dots \quad b_n),$$

$$H(z) = \mathbf{C}(z\mathbf{I} - \mathbf{A})^{-1} \mathbf{B}, \quad G(z) = \mathbf{C}(z\mathbf{I} - \mathbf{A}^M)^{-1} \mathbf{A}^{M-1} \mathbf{B}.$$

Tím je vypočtena přenosová funkce po decimaci, bez nutnosti rozkladu na kořenové činitele. Místo použité přidružené matice je ovšem možno pracovat s maticí v jiném tvaru, např. kanonickém Jordanově – umocňování Jordanových bloků je potom obdobné výpočtu koeficientů u částečných zlomků.

Uvedená metoda ovšem co do rozměrů matic a stavového vektoru zachovává řád systému, takže pokud je použita na celý přenos včetně FIR části, vzniknou ve výsledném přenosu $G(z)$ krácené nuly a póly, tvořící tak snížení řádu. Ty je možno zjistit a odstranit numericky, např. funkcí `minreal` v Matlabu. Jinou možností je algebraicky rozdělit přenos na FIR část a ryzí část a decimovat s použitím stavového popisu pouze ryzí část, přičemž zbývající decimace FIR části (včetně snížení řádu) je triviální. Rozdělení přenosu je možné i bez rozkladu na částečné zlomky:

$$H(z) = \frac{B(z)}{z^m A(z)}, \quad H_F(z) = \frac{F(z)}{z^m}, \quad H_P(z) = \frac{C(z)}{A(z)},$$

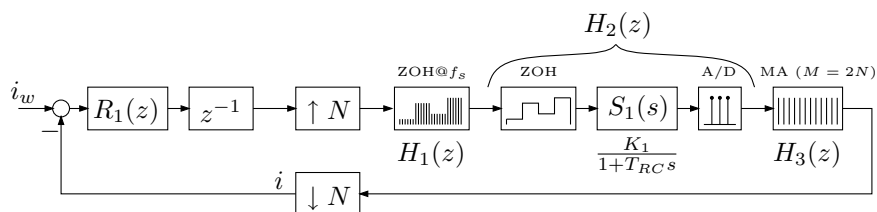
$$B(z) = F(z)A(z) + z^m C(z), \quad \deg B = \deg A + m, \quad \deg F = m, \quad \deg C = \deg A - 1.$$

Polynomiální rovnost dává soustavu $n = \deg B$ lineárních rovnic, jejíž jednotlivé rovnice odpovídají příslušným mocninám z .

Podobnou, ale méně obecnou úlohu decimace přenosu řeší funkce `d2d` v Matlabu. Ta ovšem natvrdo předpokládá, že systém obsahuje $H_1(z)$ a že jde o zadrž 0. řádu. Funkce pracuje se stavovým popisem a neprovádí redukcí řádu.

6.3.1.3. Rozbor řízeného systému

Proudová zpětnovazební smyčka z pohledu diskrétního regulátoru je zobrazena na obr. 17. PI regulátor R_1 pracuje, jak již bylo řečeno, na vzorkovací frekvenci f_{PWM} . Přechod na vyšší vzorkovací frekvenci $f_s = N f_{PWM}$ je popsán jako zadrž 0. řádu s přenosem $H_1(z)$. Akční veličina (u) dále vstupuje do spojitého systému S_1 , opět přes zadrž 0. řádu, tentokrát je již spojitou (obě zadržky jsou ve skutečnosti touž zadržky – výstupem PWM). Výstup systému (měření proudu na RC článku) je vzorkován na frekvenci f_s a vstupuje do filtru klouzavých průměrů $H_3(z)$, jehož posláním je odstranit zrcadla vyšších harmonických PWM před decimací. Následuje decimace opět na frekvenci regulátoru f_{PWM} a smyčka je uzavřena.

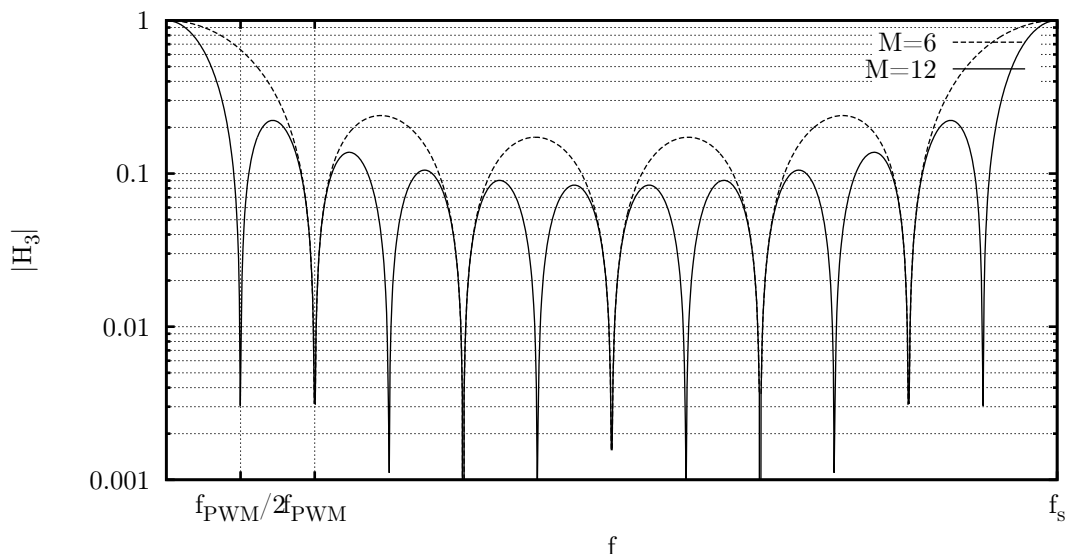


17. proudová zpětnovazební smyčka PI regulátoru

Úlohou filtru $H_3(z)$ je potlačit v měřeném signálu nežádoucí složky, které by se nepříznivě zrcadlily decimací na místa užitečného signálu a působily chybu. V [↔6.3.1.1.] bylo řečeno, že nuly filtru by se měly nalézat na harmonických signálu PWM, aby je potlačily. Jako filtr byl zvolen filtr klouzavých nevážených průměrů délky pro svou jednoduchost, neboť v implementaci je tvořen pouhým součtem nad kruhovým zásobníkem. Přenos filtru klouzavých průměrů a umístění jeho nul je

$$H_3(z) = \sum_{k=0}^{M-1} z^{-k} = \frac{z^{M-1} - 1}{z^{M-1}(z - 1)}, \quad f_n = \frac{k f_s}{M}, k \neq nM; k, n \in \mathbb{Z}, \quad H_3(e^{j2\pi \frac{fn}{f_s}}) = 0.$$

Aby filtr splňoval požadavek umístění nul, musí být jeho délka M rovna celočíselnému násobku poměru vzorkovacích frekvencí N . Nejjednodušším (nejkratším) vyhovujícím filtrem je tedy filtr délky $M = N$. V návrhu jsem ovšem zvolil filtr dvojnásobné délky $M = 2N$. Důvodem bylo dosáhnout o něco vyššího potlačení signálů v rozmezí $\langle f_{PWM}/2, f_{PWM} \rangle$ vzhledem k tomu, že po decimaci budou zrcadleny do hlavního užitečného pásma $\langle 0, f_{PWM}/2 \rangle$ (obr. 18).



18. frekvenční charakteristika filtru klouzavých průměrů

Diskrétní přenos $H_2(z)$ řízeného systému S_1 získáme diskretizací metodou ZOH:

$$H_2(z) = \mathcal{Z} \left\{ \mathcal{L}^{-1} \left\{ \frac{1}{s} S_1(s) \right\} \right\} \frac{z-1}{z} = \mathcal{Z} \left\{ \mathcal{L}^{-1} \left\{ \frac{1}{s} \frac{K_1}{1 + T_{RC}s} \right\} \right\} \frac{z-1}{z},$$

$$H_2(z) = \frac{K_d}{z-q}, \quad q = e^{-\frac{1}{T_s T_{RC}}}, K_d = K_1(1-q).$$

Nyní můžeme přistoupit k decimaci celkového přenosu $H(z) = H_1(z)H_2(z)H_3(z)$.

$$H_1(z) = \sum_{k=0}^{N-1} z^{-k}, \quad H_2(z) = \frac{K_d}{z-q}, \quad H_3(z) = \sum_{k=0}^{M-1} z^{-k},$$

$$H(z) = H_1(z)H_3(z)H_2(z) = \frac{\sum_{k=0}^{M+N-2} b_k z^k}{z^{M+N-2}} \frac{K_d}{z-q}.$$

V našem konkrétním případě $N = 6$, $M = 12$, $q \doteq 0,946$ vychází FIR část $H_F(z)$ řádu $M + N - 2 = 16$ a ryzí část $H_P(z)$ řádu 1. Po výpočtu polynomů $F(z)$ a $C(z)$ Gaussovou eliminací a následné decimaci vyšly přenosy

$$G_F(z) = \frac{f_{10}z + f_4}{z^2}; \quad (f_{16} = 0), \quad G_P(z) = \frac{c_0 q^{N-1}}{z - q^N},$$

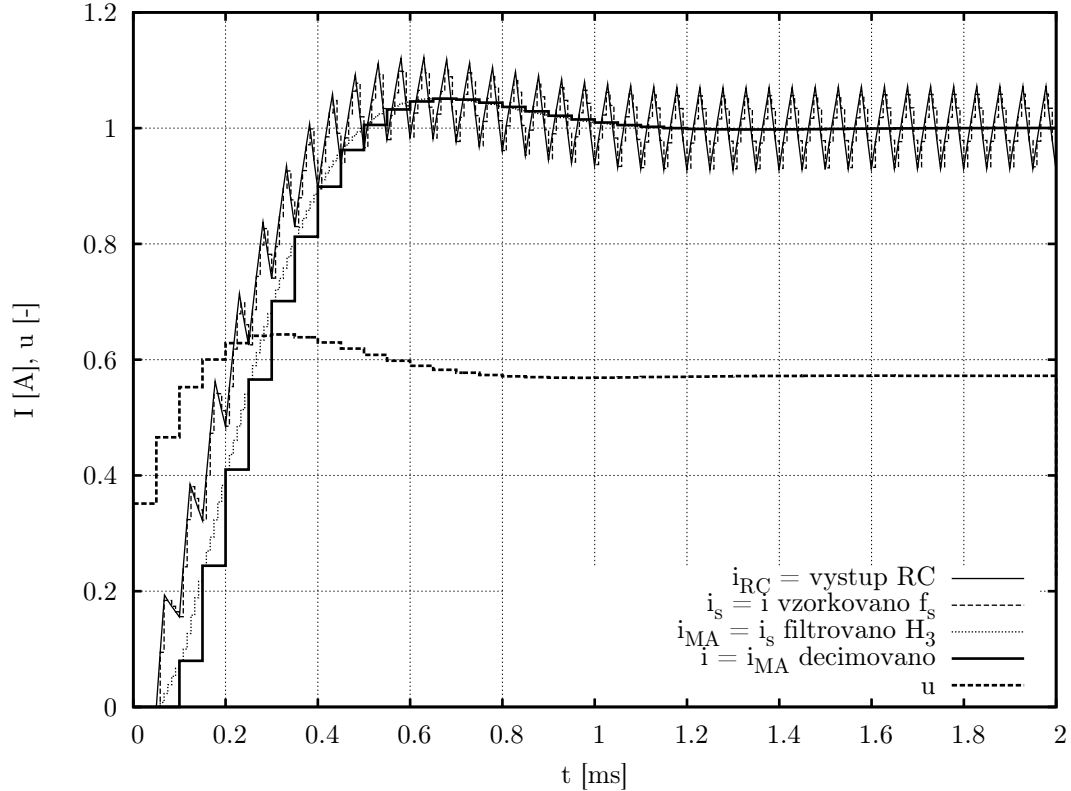
$$G(z) = G_F(z) + G_P(z) \doteq K_G \frac{(z+1)(z+0.6386)}{z^2(z-q^N)}.$$

6.3.1.4. Návrh regulátoru

Úkolem je navrhnout diskrétní PI regulátor pro systém s přenosem $G(z)$. Vzhledem k přirozenému zpoždění o 1 krok, způsobeném nenulovou dobou výpočtu, je třeba navrhnout regulátor pro řízený systém s přenosem

$$G_S(z) = \frac{1}{z}G(z) = K_G \frac{(z+1)(z+0.6386)}{z^3(z-q^N)}, \quad q^N \doteq 0,7165.$$

Přenos diskrétního PI regulátoru lze zapsat $R_1(z) = K(z-n)/(z-1)$, parametry jsou nula přenosu n a celkové zesílení K . Regulátor byl nastaven zkusmo s použitím nástroje Matlab SISO Design Tool (`sisotool`). Cílem návrhu bylo udržet přechod v rozumných mezích (hlavním úkolem regulátoru je zabezpečit omezení proudu). Výsledné parametry regulátoru jsou $n = 0,674$, $K = 0,6/G_S(1)$ ¹³. Přechodovou charakteristiku uzavřené smyčky ukazuje obr. 19. Přechod odezvy činí 5,74 % a nastává v čase $0,65 \text{ ms} = 13T_{PWM}$.

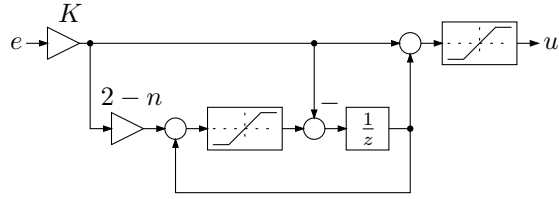


19. přechodová charakteristika uzavřené proudové smyčky ($i_w = 1 \text{ A}$)

6.3.1.5. Realizace

Regulátor je obsažen v softwaru desky SPEJBL-ARM. Výpočty byly naprogramovány v celočíselné 32-bitové aritmetice tak, aby byl dynamický rozsah celých čísel využit co nejvíce, s tím, že jediné použité dělení (měřítko akční veličiny na výstupu, kde je hodnota přepočtena na rozsah periody čítače PWM) je mocninou 2, konkrétně 2^{15} , takže celá implementace obsahuje pouze dvě 32-bitová násobení, jeden bitový posun (dělení) a několik sčítání a odčítání.

¹³ $G_S(1)$ značí stejnosměrný přenos řízené soustavy, tj. fyzikálně vodivost vinutí

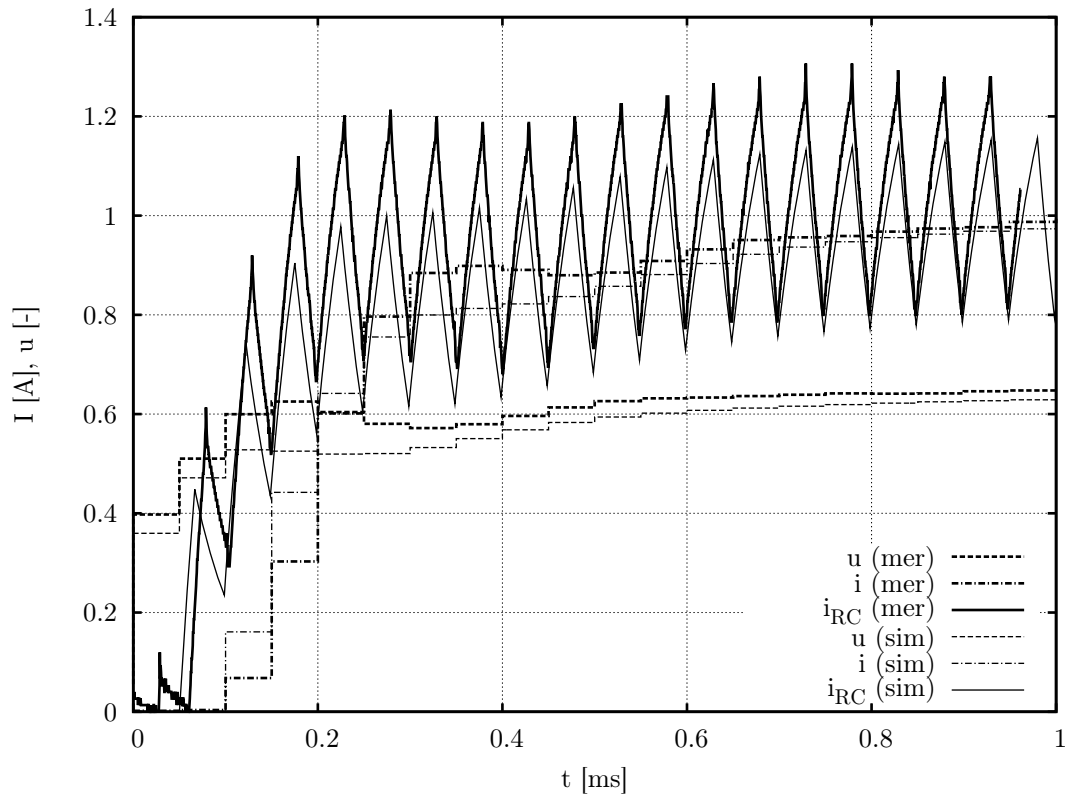


20. realizace PI regulátoru proudu

Proudové omezení veličiny i_w bylo pro jistotu vloženo i na vstup regulátoru, aby nemohlo dojít omylem v návrhu vnější smyčky k nastavení proudu chybně mimo povolený rozsah. Maximální přípustná hodnota proudu byla nastavena na hodnotu $|i_{max}| = 1$ A.

Navržená elektronika [↔4.3.2.] měří proud pouze unipolárně, v absolutní hodnotě. Informace je tedy neúplná. V regulátoru jsem tento nedostatek obešel tak, že natvrdo předpokládám, že znaménko proudu je shodné se znaménkem akčního napětí. To samozřejmě nemusí být pravda, jednak vinou přechodového děje RC článku, zasahujícího do příští periody PWM, kdy už může mít proud opačné znaménko, dále pak tím, že znaménko proudu a budícího napětí nemusí být shodné.

Pro zamezení windupu (přeplnění integrátoru v případě nasycení akční veličiny u) jsem použil ošetření *tvrdým anti-windupem*, více viz [↔6.3.2.4.]. V implementaci jsem je realizoval pomocí dvou sčítání s kontrolou přetečení a jedné paměťové buňky, jak je naznačeno na obr. 20. Stav integrátoru je držen v mezích omezení akční veličiny, upravených současnou hodnotou proporcionální složky. Nahradíme-li omezující bloky v diagramu krátkým spojením, bude mít výsledný obvod požadovaný přenos $R(z) = K(z - n)/(z - 1)$.



21. srovnání měřené a simulované přechodové charakteristiky proudového regulátoru ($i_w = 1$ A)

Graf na obr. 21 ukazuje srovnání měřené a simulované odezvy uzavřené smyčky na jednotkový skok referenční veličiny. V měření i simulaci byl řízen systém s parametry změněnými oproti návrhu, časová konstanta T_{RC} byla v pokusném systému asi $3\times$ menší, než v návrhu a konečném výrobku. Rovněž statické zesílení (vodivost vinutí) se mírně lišilo. Průběh pilotního napětí na RC článku měření proudu byl změřen digitálním osciloskopem Tektronix TDS210 v režimu průměrování $16\times$, zbylé hodnoty byly přeneseny z desky SPEJBL-ARM po síti CAN. (Simulovaná odezva systému s časovou konstantou T_{RC} v souladu s návrhem regulátoru byla na obr. 19.)

6.3.2. Polohový PID regulátor

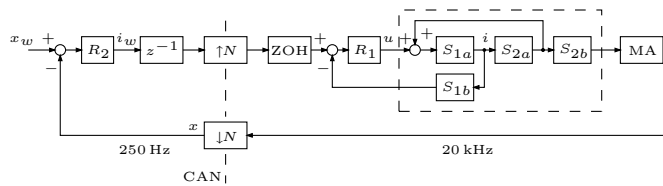
Polohový regulátor R_2 je regulátorem vnější smyčky kaskády. Měřenou a referenční veličinou je poloha (úhel) x , akční veličinou proud i_w . Aby bylo možné v budoucnu vnější SISO regulátor nahradit obecným MIMO regulátorem a ověřit uzavření smyčky přes síť CAN, byl tento regulátor realizován v softwaru hlavního řídicího počítače. Vzorkovací frekvence sítě, tudíž i měřené a akční veličiny regulátoru, je $f_{CAN} = 250$ Hz. Vnitřní smyčka běží na frekvenci $f_{PWM} = 80f_{CAN}$, výstup (potenciometr) je možno rovněž vzorkovat na frekvenci f_{PWM} – ve vnější smyčce se tedy opět vyskytuje změna vzorkovací frekvence a přenos je decimován, obdobně jako v [↔6.3.1.3.].

Vnořená regulační smyčka nemění řád astatismu řízené soustavy oproti původní napěťově řízené, popsané v [↔6.2.1.]. Dle závěrů učiněných při návrhu P regulátoru je pro potlačení konstantní poruchy na vstupu motoru (způsobené zátěžovým momentem) třeba, aby regulátor obsahoval astatismus alespoň 1. řádu (integrační složku). Jak se v rozboru i při pokusu ukázalo, je rovněž přítomnost diferenční složky nezbytná, neboť zpoždění v řízeném systému, způsobená filtrací a přenosem po síti, způsobují značný fázový posun. Zvoleným regulátorem je tudíž PID.

6.3.2.1. Rozbor

Změna vzorkovací frekvence probíhá uvnitř softwaru desky SPEJBL-ARM. Na straně požadovaného proudu i_w je tvořena zádrž. Byla použita zádrž 0. řádu, ovšem výhledově by mohla být přínosem zádrž 1. řádu, vzhledem k tomu, že vzorkování f_{CAN} je poměrně pomalé. Měřená poloha x je před decimací filtrována opět klouzavými průměry, v tomto případě byl zvolen filtr délky $M = N = 80$ jako kompromis mezi využitím všech měřených dat a co nejmenším zpožděním FIR filtru (v tomto zvláštním případě se klouzavý průměr rovná průměru blokovému). Vzorkování obou smyček je asynchronní, což způsobuje chybu časování (jitter) nejvýše $1/80$ periody vnější smyčky.

Schéma řízeného systému z pohledu regulátoru R_2 je na obr. 22. Dynamika řízeného systému obsahuje zádrž, PI regulátor proudu, RC článek, mechanickou setrvačnost, astatismus polohy, průměrovací filtr. To vše je decimováno a přidáno je přirozené zpoždění o jeden krok periody na sběrnici CAN, v němž proběhne výpočet regulátoru v hlavním řídicím počítači a přenos zpráv s měřenými a akčními veličinami. Jednou z možností modelování řízeného systému by bylo stanovit dílčí přenosy, tedy zejména upřesnit mechanickou časovou konstantu a provést decimaci. Vyjde přenos zbytečně vysokého řádu, obsahující póly s velmi rozdílnou dynamikou – pomalou mechanickou setrvačností oproti rychlým dějům v proudové smyčce. Je vhodné model zjednodušit a popsat přenosem nižšího řádu.



22. polohová zpětnovazební smyčka PID regulátoru

Druhou možností je identifikovat celý řízený systém z naměřených dat. Vzhledem k tomu, že navrhovaný polohový PID regulátor měl být opět nastaven se shodnými parametry pro všechny klouby a okamžiky chůze, mění se dynamika mechanické části řízeného systému ve velkém rozpětí. Proto nemá v modelování smysl popisovat podrobně prvky dynamiky, jejichž význam je vůči těmto změnám zanedbatelný.

6.3.2.2. Identifikace řízeného podsystému

Úkolem identifikace je stanovit přenos řízeného systému $P(z)$ ze vstupu i_w na výstup x . Za tímto účelem je systém třeba vybudit vhodným signálem $i_w[k]$ a provést měření výstupu $x[k]$. Při buzení systému pseudonáhodnou dvojkovou posloupností jsem nedosáhl příliš dobré shody modelovaného výstupu s měřeným. Na vině mohly být, kromě špatné volby posloupností, též nelinearity uvnitř systému, například omezení proudu, k němuž docházelo. Naproti tomu se osvědčilo buzení sinusovými signály. Pro identifikaci jsem použil algoritmy odhadu parametrických modelů nástroje Matlab System Identification Tool.

Při identifikačních pokusech byl motor připevněn za hřídel a setrvačná hmota tak odpovídala vlastní setrvačnosti těla motoru. Opět se jedná o setrvačnost pod dolní mezí skutečných setrvačností uvnitř robota. Identifikaci jsem prováděl z dvojích dat. Nejprve jsem vybudil systém rozmítaným signálem (chirp). Použil jsem exponenciálně rozmítanou frekvenci v rozsahu $\langle f_0, f_1 \rangle$ Hz podle funkce `chirp(t, f0, t(end), f1, 'logarithmic')` Matlabu:

$$A = 0,2 \text{ A}, k = 0 \dots N, N = f_s 120 \text{ s}, f_0 = 0,2 \text{ Hz}, f_1 = 125 \text{ Hz}, \quad i_w[k] = A \cos 2\pi\phi[k],$$

$$\phi[k] = \frac{N}{\ln \frac{f_1}{f_0}} f_0 \left(\left(\frac{f_1}{f_0} \right)^{\frac{k}{N}} - 1 \right).$$

Měřená data jsem posunul o 1 vzorek (vyrovnání přirozeného zpoždění výpočtu a síťového provozu) a odstranil střední hodnotu a trend. Rušivý trend se objevil v oblasti vyšších frekvencí tím, jak nahodile zabíraly nelineární složky tření motoru. Trend jsem odhadl jako signál, filtrovaný dolní propustí FIR 4000. řádu se zlomovou frekvencí $2 \cdot 10^{-4} f_{CAN}$ a nulovou fází.

Z bloku těchto dat v časové oblasti byly identifikovány modely ve stavovém popisu metodou N4SID ([15], 10.6) a přenosové modely typu OE (output error) metodou PEM ([15], 7.2). Z několika identifikovaných modelů řádu 2...5 jsem ponechal stavový model 2. řádu `n4s2` a OE model 2. řádu se zpožděním o 2 kroky `oe222`. Při dalším zvyšování řádu již nedocházelo k významnému snížení chyby predikce. Přenosy obou modelů vyšly (po obohacení zpožděním o 1 krok) ve tvaru

$$P(z) = K_P \frac{z - n}{z(z - 1)(z - q)},$$

v modelu **n4s2** s nulou $n = 1,783$ a pólem $q = 0,8114$, v modelu **oe222** $n = 3,584$ a $q = 0,7958$.

Pro ověření jsem se rozhodl provést ještě identifikaci parametrického modelu z dat ve frekvenční oblasti. Pro ten účel jsem sejmul ustálené odezvy na sinusové buzení o frekvencích

$$f \in \{2, 7, 10, 15, 18, 20, 22, 25, 30, 40\} \text{ Hz.}$$

Z každého ustáleného stavu bylo naměřeno 1250 vzorků, tj. 5 s úsek. Úsek byl ještě zkrácen tak, aby obsahoval pokudmožno celý počet period budícího signálu. Komplexní přenos v bodě f byl poté stanoven aritmetickým průměrem

$$\vartheta = 2\pi \frac{f}{f_{CAN}}, i_w[k] = A \cos \vartheta k, \quad \hat{P}(e^{j\vartheta}) = \frac{1}{N} \sum_{k=0}^N e^{j\pi\vartheta} x[k].$$

Ze získaných bodů frekvenční charakteristiky byl opět odhadnut parametrický model typu OE. Vybral jsem model 2. řádu se zpožděním o 2 kroky **oe222f**. Frekvenční charakteristika již v sobě zahrnuje fázový posun způsobený přirozeným zpožděním výpočtu a sítě, takže není k výslednému modelu přidáváno. Přenos modelu **oe222f** vyšel opět v téže podobě, s nulou $n = 2,02$ a pólem $q = 0,7592$. U všech tří modelů jsem poopravil polohu pólu integrátoru, aby ležela přesně v $z = 1$.

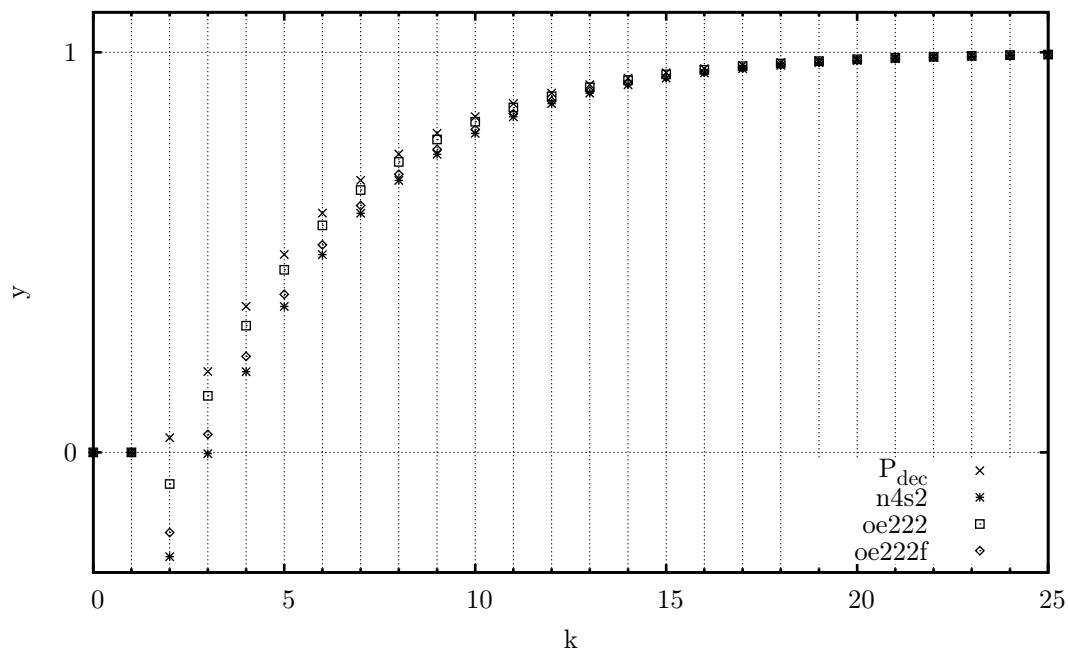
Srovnajme získané modely s přenosem vypočítaným decimací podle [←6.3.1.2.1.]. Dynamiku vnitřní smyčky jsem zanedbal jako příliš rychlou a nahradil ji statickým zesílením. Zjednodušený přenos systému na vzorkovací frekvenci f_{PWM} je

$$H(z) = H_1(z)H_2(z)H_3(z), \quad H_1(z) = H_3(z) = \sum_{k=0}^{N-1} z^{-k}, \quad H_2(z) = K_s \frac{z + n_s}{(z - 1)(z - q_s)},$$

kde $N = 80$ a přenos $H_2(z)$ vznikl diskretizací přenosu $K/s(T_m s + 1)$. Výsledný přenos po decimaci vychází ve tvaru

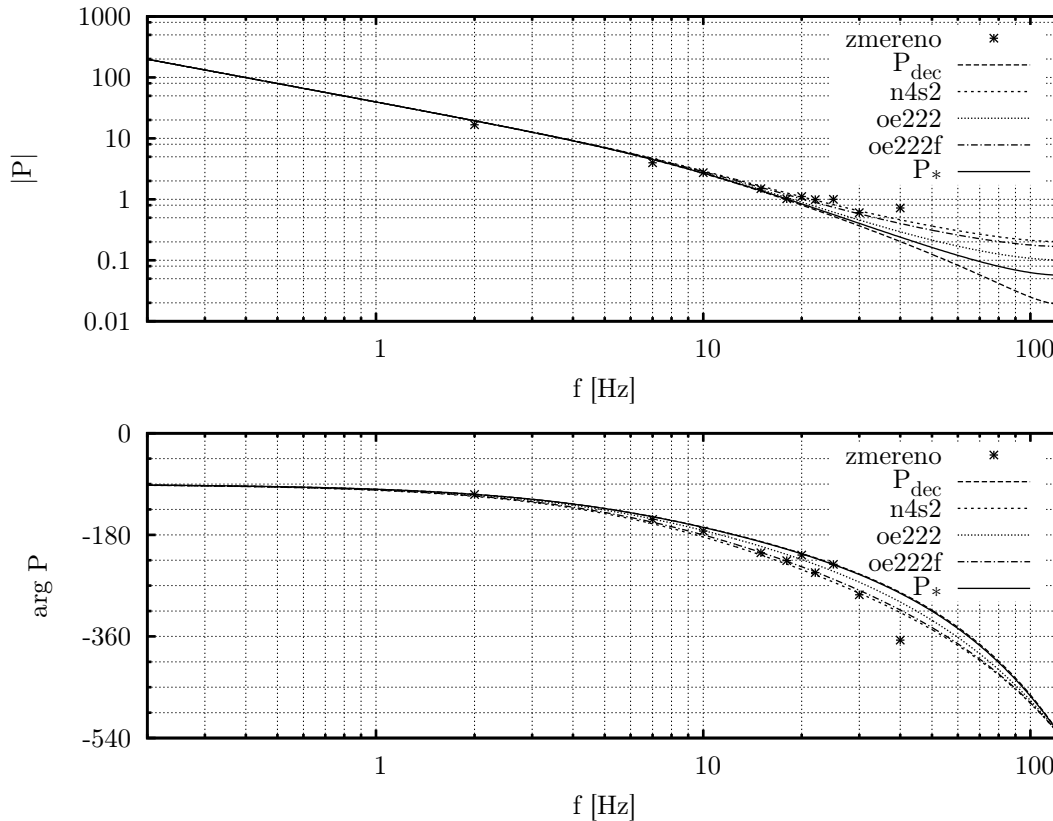
$$P_{dec}(z) = K_G \frac{(z + n_1)(z + n_2)}{z^2(z - 1)(z - q)},$$

s nulami na záporné reálné poloose, minimálněfázovou $n_1 = 0,2481$ a neminimálněfázovou $n_2 = 3,463$. Pól q odpovídá mechanické časové konstantě motoru.



23. impulsová charakteristika modelů $P(z)$

V čem se jednotlivé modely liší a co mají společného? Uvažujme ve všech modelech shodný pól $q = 0,7958$ vzatý z modelu *oe222* a shodné zesílení. Impulsovou odezvu modelů ukazuje graf na obr. 23, frekvenční charakteristiku obr. 24. Impulsová odezva modelů je tvořena exponenciálním náběhem, tvořeným odezvou pólu q integrovanou astatismem. Lze říci, že čelo této odezvy je zpožděno o 3 kroky. V tom se modely shodují, včetně modelu $P_{dec}(z)$, který má sice v jmenovateli jedno zpoždění navíc, ale to je kompenzováno dominantním koeficientem v čitateli. Modely se liší nejvíce v obsahu 3. vzorku $h[2]$ impulsové odezvy. Tři identifikované modely v něm mají záporný impuls, způsobený neminimálněfázovou nulou. Model P_{dec} v něm má naopak nepatrný impuls kladný.



24. frekvenční charakteristika modelů $P(z)$

Záporný impuls předcházející čelu exponenciály není strukturou řízeného systému opodstatněn. Přítomností nuly $n > 1$ je tedy zřejmě vina numerická chyba identifikačních algoritmů. Nula by patrně měla ležet až v nekonečnu, ale v odhadu tak daleko nedoputovala. Vzhledem k tomu, že nula se nalézá na nulové frekvenci, kde zároveň účinkuje astatismus systému, a navíc poměrně daleko od jednotkového kruhu, nelze její přítomnost z měřených dat dobře ověřit. Pravděpodobnější je výskyt nul na záporné poloose v modelu P_{dec} , který mírně tlumí frekvenční charakteristiku systému v blízkosti $f_{CAN}/2$. Pro účely řízení servomotoru není ovšem tento vliv podstatný. Kromě toho je třeba mít na paměti, že tento model vznikl výpočtem za zjednodušujícího předpokladu statického přenosu uzavřené vnitřní smyčky R_1, S_1 . Proto jsem se rozhodl případné nuly systému zanedbat s tím, že impulsovou charakteristikou je o 3 kroky zpožděná exponenciála, tvořená pólem s převládající časovou konstantou a astatismem systému. 3 kroky zpoždění obsahují zádrž nultého řádu při změně vzorkovací frekvence ($\sim 1/2$ kroku zpoždění), průměrovací filtr (rovněž $\sim 1/2$ kroku zpoždění), 1 krok přirozeného zpoždění na síti CAN a zbývající krok odpovídá zpoždění přísně ryziho systému. Výsledný model systému, řízeného vnější smyčkou polohové zpětné vazby, je:

$$P(z) = \frac{K_P}{z(z-1)(z-q)}, \quad q = 0,7958.$$

Frekvenční charakteristika $P(z)$ je vyznačena na obr. 24 plnou čarou. Hlavní mechanická časová konstanta vychází ze vztahu $e^{-1/f_{CAN}T_m} = 0,7958$ rovna $T_m = 17,5 \cdot 10^{-3}$ s. Je tedy o něco větší u setrvačnosti těla motoru než u setrvačnosti ramene N5a. Bylo by přínosné provést identifikaci s alespoň dvěma různými známými setrvačnými hmotami, připojenými na hřídel motoru (např. poprvé s žádnou a podruhé se známou setrvačností). Z naměřených

časových konstant by pak bylo možné vypočítat vlastní setrvačnost uvnitř motoru a součinitel lineárního tření, a následně časovou konstantu mechanismu pro libovolnou známou setrvačnou zátěž připojenou na hřídel.

6.3.2.3. Návrh regulátoru

Přenos použitého diskrétního PID regulátoru je

$$R_2(z) = \frac{d_2 z^2 + d_1 z + d_0}{z(z-1)} = r_P + r_I \frac{1}{z-1} + r_D \frac{z-1}{z},$$

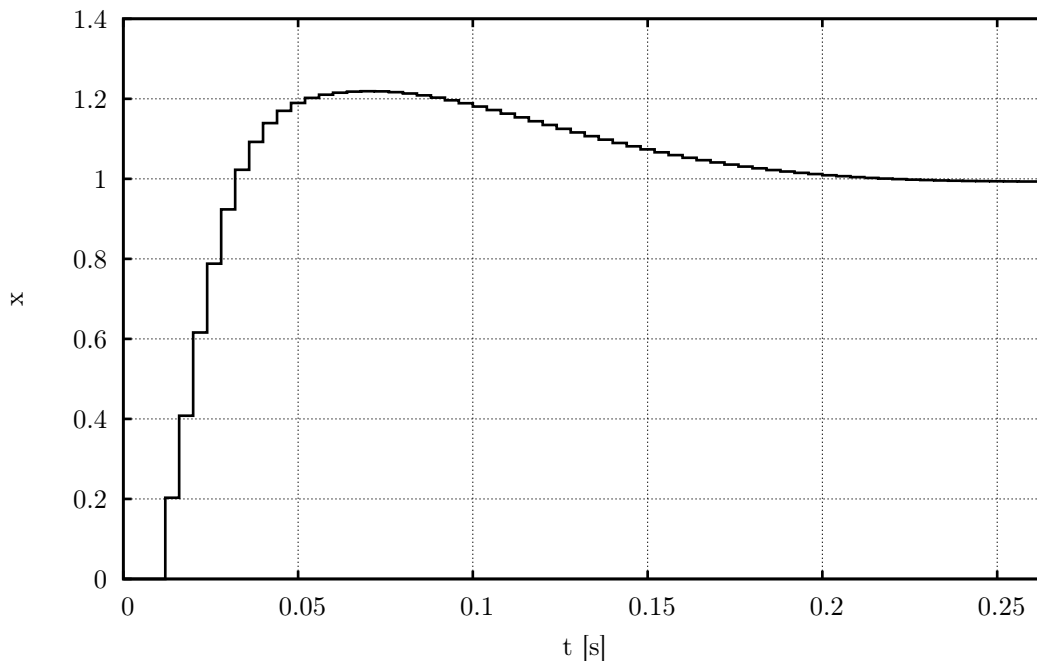
kde parametry r_P, r_I, r_D odpovídají konstantám proporční, integrační a derivační složky regulátoru s aproximací integrátoru o krok zpóźděným sumátorem a aproximací derivátoru zpětnou diferencí. Zesílení K a časovým konstantám derivační a integrační složky T_D, T_I [s.rad⁻¹] spojitěho PID regulátoru

$$R(s) = K \frac{(T_D s + 1)(T_I s + 1)}{T_I s}$$

odpovídají v přiblížení $T_{I,D} \gg 1/f_{CAN}, T_I \gg T_D$ parametry

$$r_P = K, \quad r_I = K \frac{1}{T_I f_{CAN}}, \quad r_D = K T_D f_{CAN}.$$

Kritická frekvence systému $P(z)$ je $\omega_k = 78,1 \text{ rad.s}^{-1} = 12,4 \text{ Hz}$. Derivační frekvenci regulátoru jsem umístil do její blízkosti, $\omega_D = 63,5 \text{ rad.s}^{-1}$. Integrační frekvenci jsem zvolil 5× nižší, $\omega_I = 12,6 \text{ rad.s}^{-1}$. Zesílení regulátoru jsem ladil ručně s ohledem na omezení akční veličiny a typické maximální velikosti vyskytnuvších se skoků a na výsledný překmit. Zvolil jsem $K = 41,1 \cdot 10^{-3} / K_P$. Výsledná přechodová charakteristika je na obr. , překmit činí 22 % (v lineární pracovní oblasti – vlivem windupu může být při nasycení akční veličiny větší), k ustálení s chybou pod 1 % dojde za 0,2 s.



25. přechodová charakteristika uzavřené polohové smyčky

6.3.2.4. Realizace

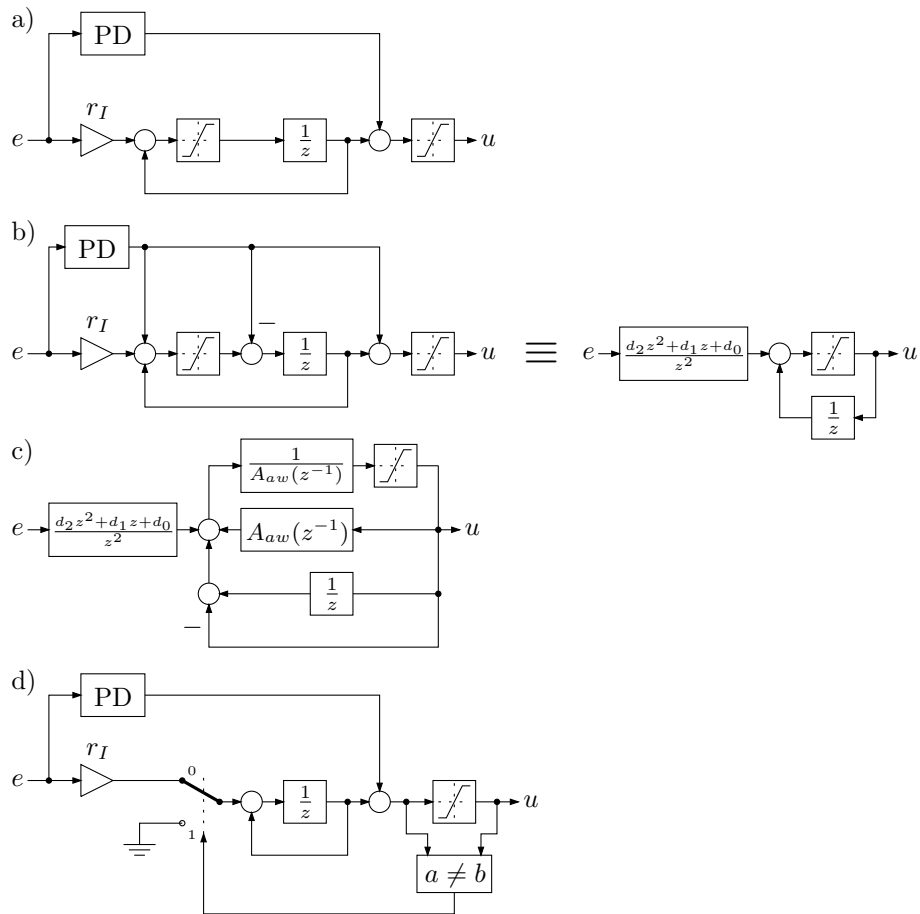
Polohový PID regulátor pro všechny servomotory robota je obsažen v softwaru hlavního řídicího počítače. K usnadnění úprav a změn v návrhu bylo použito aritmetiky v plovoucí řádové čárce.

Při vylepšování PID regulátoru, vedeného zejména snahou odstranit závadné kmitání regulátoru [↔6.3.3.2.], jsem vyzkoušel několik vylepšení PID regulátoru:

- filtraci derivační složky – nahrazení zpětné diference $(z - 1)/z$ filtrovanou diferencí $(1 - f)(z - 1)/(z - f)$, $f \rightarrow 0$ nepřineslo žádnou znatelnou změnu; vzorkovací frekvence f_{CAN} je poměrně nízká na to, aby filtrace měla výraznější vliv;
- statickou nelinearitu na vstupu – snažil jsem se zlepšit chování regulátoru v ustáleném stavu vkládáním statických nelinearit, jak radí např. [↔9.1.16.]; přítomnost hladkých funkcí ani umělých pásem necitlivosti nepřineslo žádné zlepšení;
- vyloučení referenčního signálu z derivační větve;
- několik variant omezení windupu.

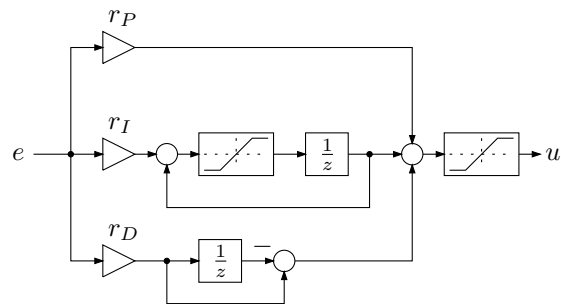
Zdaleka nejzávažnějším se ukázalo být řešení ochrany proti windupu, tj. přeplnění integrátoru nad mez danou omezením akční veličiny (více viz [12]). Různé techniky anti-windupu se velmi lišily v účinnosti potlačení windupu na jedné straně a naproti tomu v citlivosti na poruchy. Byly prozkoumány následující metody potlačení windupu:

- *Měkký anti-windup*, obr. 26a. V této metodě je mez naplnění integrátoru dána pevně, zcela bez ohledu na vstup a stav zbytku regulátoru. Vyznačuje se nejnižší účinností potlačení windupu a velkou odolností vůči poruchám, neboť vstupní poruchy nijak tvrdě nezasahují do mezí integrátoru a účinkují na integrátor jen s pomalou dynamikou integrační větve. Tento způsob anti-windupu byl ponechán v konečné verzi regulátoru. Odpovídá analogovému řešení, kde dochází k nasycení integračního operačního zesilovače.
- *Tvrký anti-windup*, obr. 26b. Tento způsob je protikladem měkkého anti-windupu – aktuální omezení hodnoty akumulované v integrátoru se řídí situací na výstupu, tj. je dáno kromě obsahu integrátoru též vstupní odchylkou a stavem zbytku regulátoru (dynamiky diferenčního členu). Jakmile celkový výstup přesáhne omezení akční veličiny, je obsah integrátoru omezen. Tento anti-windup je velice účinný, ale zároveň citlivý na poruchy (šum na měřené odchylce). Byl použit v prvotním návrhu a následně zavrhnut, protože způsoboval nepochopitelné a do značné míry náhodné poryvy na výstupu, kdy např. rázem, bez zjevné vnější příčiny, hodnota výstupu „ustřelila“ a poněkud nepravidelně se vracela do ustáleného stavu.
- *Dynamický anti-windup*, obr. 26c je vlastně zobecněním předchozího přístupu. Hodnota akumulovaná v integrátoru je ovlivňována vstupem a stavem ostatních členů regulátoru, ovšem nikoli přímým dosazením, ale pomocí dynamického systému – pozorovatele odchylky, způsobené nasycením akční veličiny. Dynamika pozorovatele je dána polynomem $A_{aw}(z^{-1})$ ([11]). Tvrký anti-windup je zvláštním případem dynamického anti-windupu, kdy $A_{aw} = 1$ (pozorovatel typu „deadbeat“).
- *Podmínkový anti-windup*, obr. 26d jsem použil v počátcích návrhu jako heuristické a teoreticky neopodstatněné řešení. Toto řešení fungovalo poměrně dobře, netrpělo tolik na poruchy jako tvrdý anti-windup a nezpůsobovalo částečný windup, jako měkký anti-windup. Pro jeho teoretickou nepodloženost a s tím spojenou nevyzpytatelnost v neznámých situacích jsem tuto metodu opustil a uvádím ji jen pro úplnost.



26. různé způsoby ochrany proti windupu

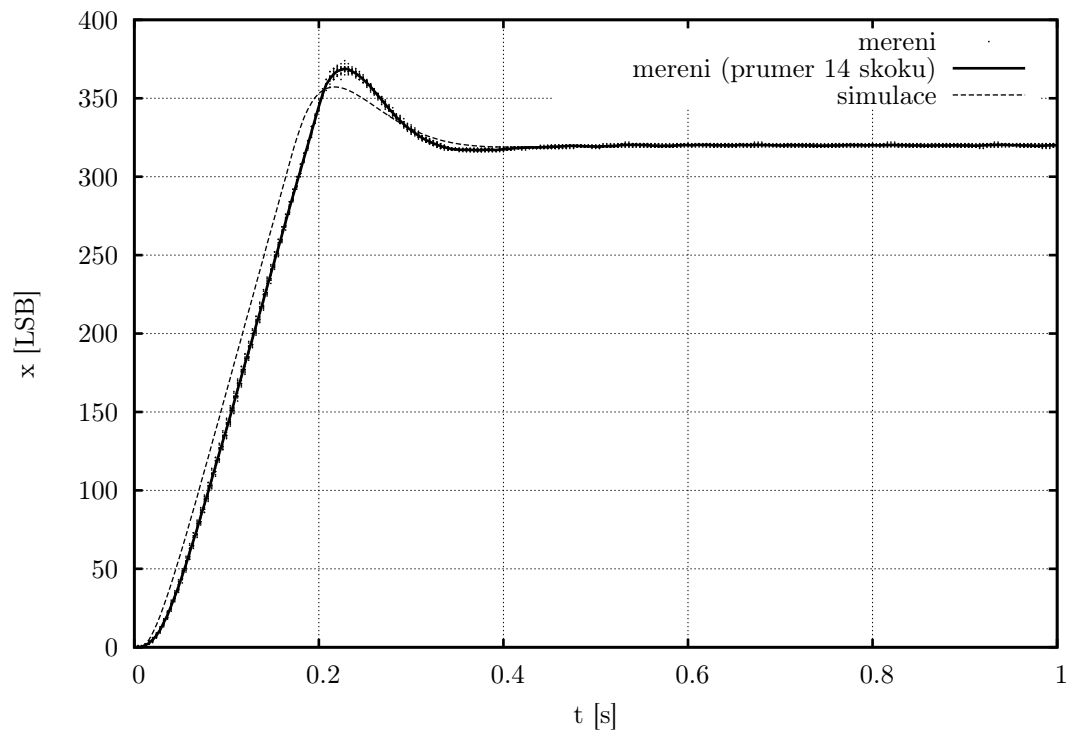
Výsledné schéma použitého diskrétního PID regulátoru je na obr. 27.



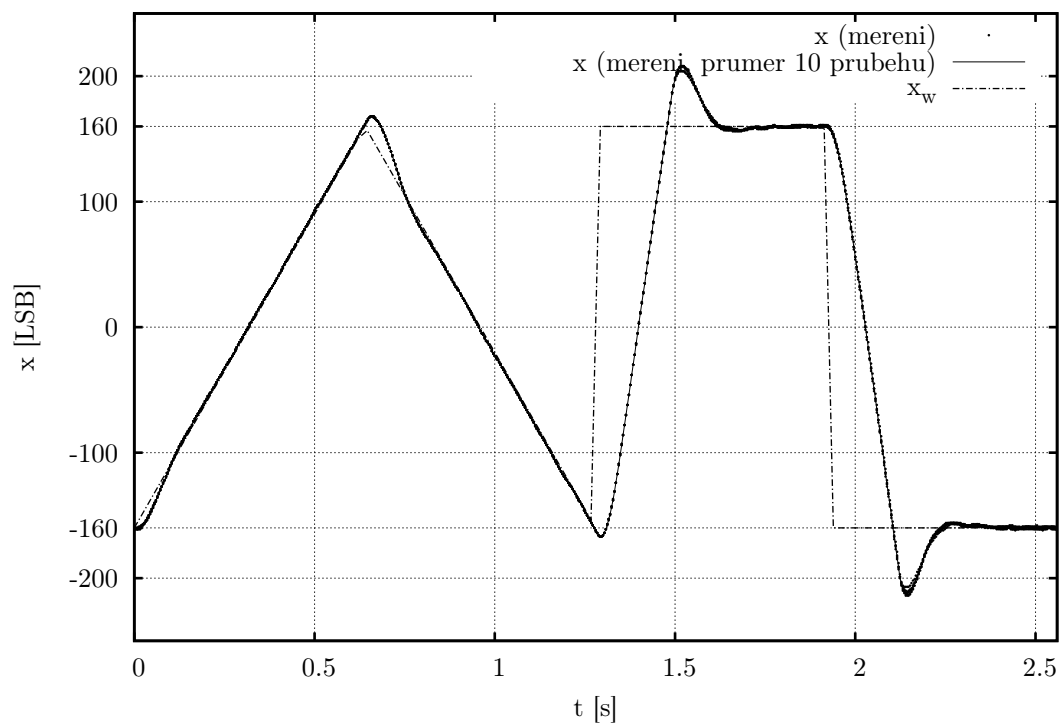
27. realizace polohového PID regulátoru

6.3.3. Výsledky

Na obr. 28 je změřená přechodová charakteristika uzavřené smyčky PID regulátoru v porovnání se simulací. Velikost skoku činí $320 \text{ LSB} \doteq 0,38\pi \text{ rad}$. Ve skutečnosti i v simulaci je použit měkký anti-windup, takže překmit je větší než v lineárním rozboru. Měření sledování ramp a skoků je na obr. 29.



28. odezva uzavřené polohové smyčky na skok



29. měřená odezva uzavřené polohové smyčky na skoky a rampy

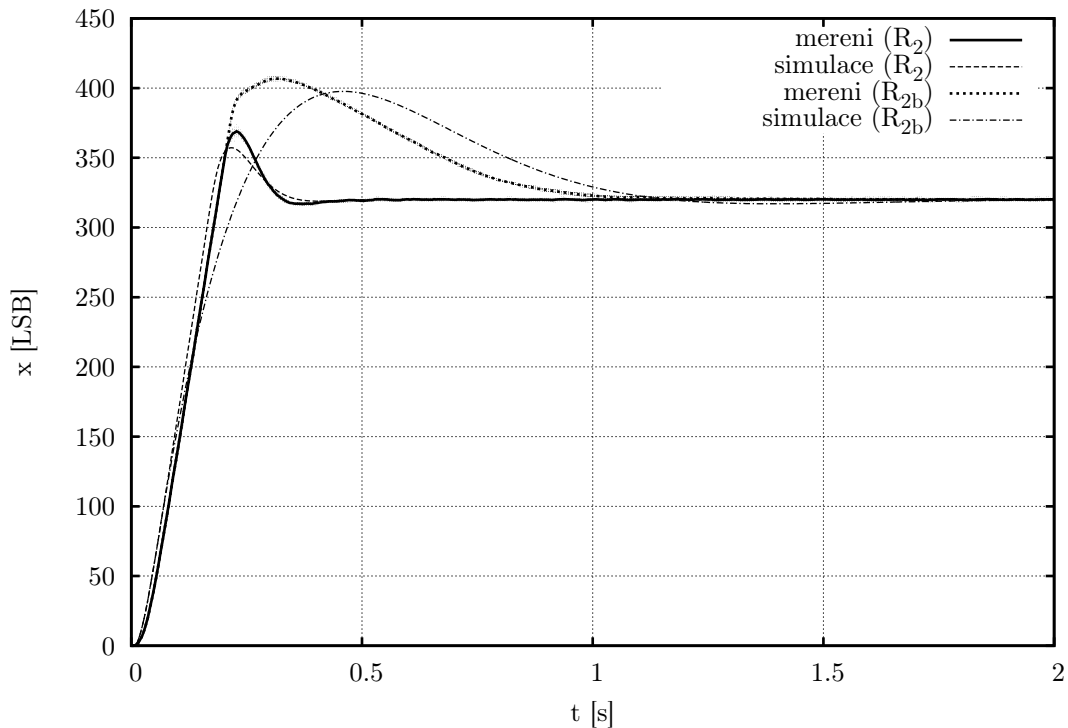
6.3.3.1. Stabilita regulátoru uvnitř robota

Polohový PID regulátor byl navržen na základě modelu, jehož převažující časová konstanta $T_m = 17,5 \cdot 10^{-3}$ s byla dána pouze vlastní setrvačností motoru. Uvnitř robota musí servomechanismus pracovat s velkým rozpětím setrvačných hmot, často mnohonásobně větších. Navržený regulátor $R_2(z)$ ztrácel stabilitu, když setrvačnost ve smyčce příliš vzrostla, jmenovitě když robot vykročil a zvedl nohu. Proto jsem regulátor přeladil, aby byl robustnější, za cenu prodloužení přechodového děje. Přeladění jsem provedl zkusmo na připojeném robotu a následně zkontroloval výsledek zpětným rozborem. Parametry regulátoru před přeladěním (R_2) a po přeladění (R_{2b}) jsou:

$$R_2 : \quad \omega_D = 63,5 \text{ rad.s}^{-1}, \omega_I = 12,6 \text{ rad.s}^{-1}, K = 41,1 \cdot 10^{-3} / K_P$$

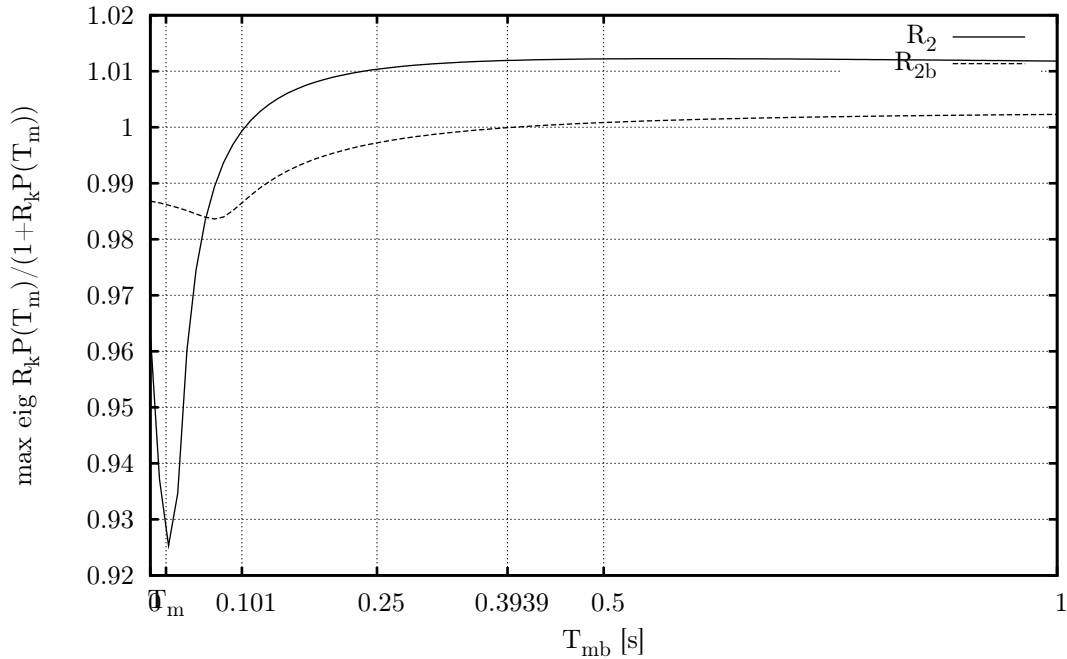
$$R_{2b} : \quad \omega_D = 25,0 \text{ rad.s}^{-1}, \omega_I = 3,12 \text{ rad.s}^{-1}, K = 7,21 \cdot 10^{-3} / K_P.$$

Srovnání přechodových charakteristik uzavřené smyčky s původním regulátorem R_2 a přeladěným R_{2b} v měření i simulaci je na obr. 30, obojí pro původní systém $P(z)$.



30. srovnání odezvy polohové smyčky na skok pro oba regulátory

Abych získal představu o robustnosti regulátoru, která byla získána přeladěním, vyhodnotil jsem numericky interval stability v parametru T_m . Tento interval udává, pro jaké časové konstanty $T_{mb} = \alpha T_m$, tedy pro jaké násobky α původní setrvačné hmoty, bude uzavřená smyčka s daným regulátorem stabilní. Podmínkou stability diskrétní uzavřené smyčky je, aby největší z absolutních hodnot pólů uzavřené smyčky byla menší než 1. Grafy na obr. 31 ukazují převažující absolutní hodnoty pólů uzavřené smyčky pro R_2 i R_{2b} . Regulátor R_2 je stabilní při setrvačnostech v rozsahu $\alpha < 5,83$, regulátor R_{2b} je stabilní v rozsahu $\alpha < 22,7$.

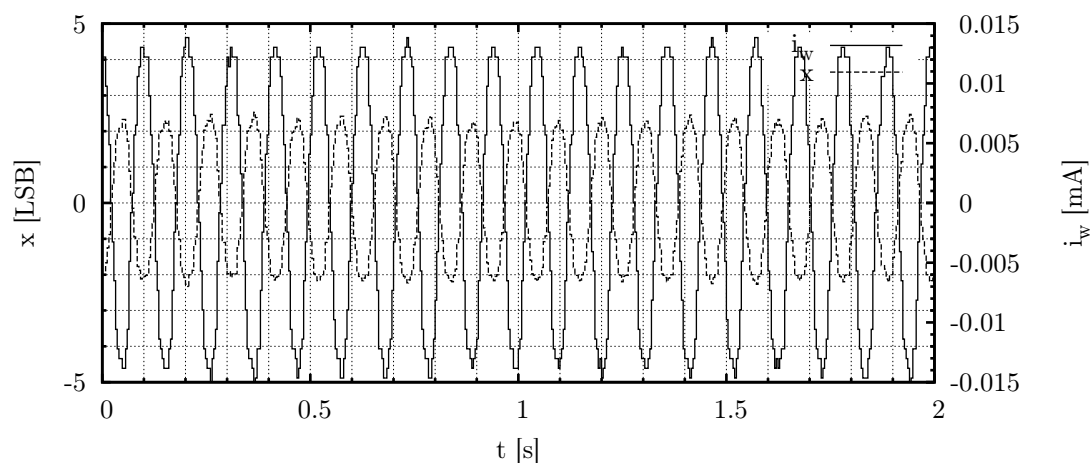


31. interval stability regulátorů R_2, R_{2b} podle T_m

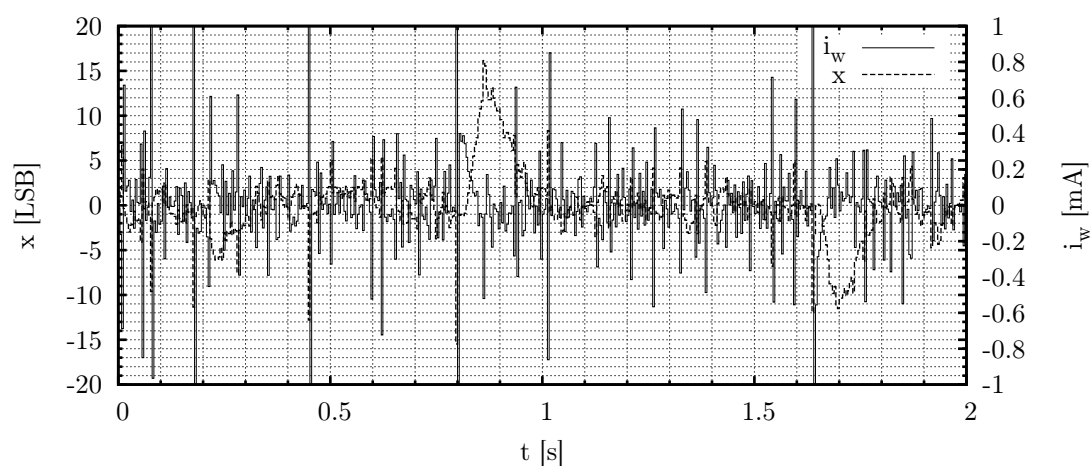
6.3.3.2. Rušivé kmitání

Nejzávažnější vadou navrženého kaskádního PID-PI regulátoru se ukázalo být rušivé kmitání, k němuž dochází nejčastěji v ustáleném stavu, ale bývá někdy i superponováno na jinak očekávaných průbězích v polohové smyčce. Stává se například, že smyčka PID regulátoru, držící konstantní polohu, začne mírně kmitat. Amplituda kmitů v místě měřené veličiny, polohy x , není někdy nijak zvlášť výrazná, často mívá rozkmit v rozmezí ± 1 LSB. Výrazněji je vidět na akční veličině i_w (obr. 32). Kmity mají podobu buď frekvenčně i amplitudově velmi pravidelných periodických kmitů o frekvenci blízké kritické frekvenci přenosu systému, tj. $\sim 12,4$ Hz. Jindy se místo pravidelných kmitů objeví zcela chaotický šum, rovněž ustáleného rozptylu (obr. 33). Jak jsem s překvapením pozoroval, systém přechází mezi třemi různými, většinou zřetelně časově i druhově oddělenými stavy:

- nulová regulační odchylka – žádoucí stav, kdy žádný z regulátorů nepůsobí; stačí ovšem nepatrná porucha a systém stav opustí, ovšem často je schopen se do něj i vrátit;
- pravidelné kmity – systém ustáleně, téměř harmonicky kmitá na své kritické frekvenci;
- chaos – systém generuje téměř dokonale bílý šum.

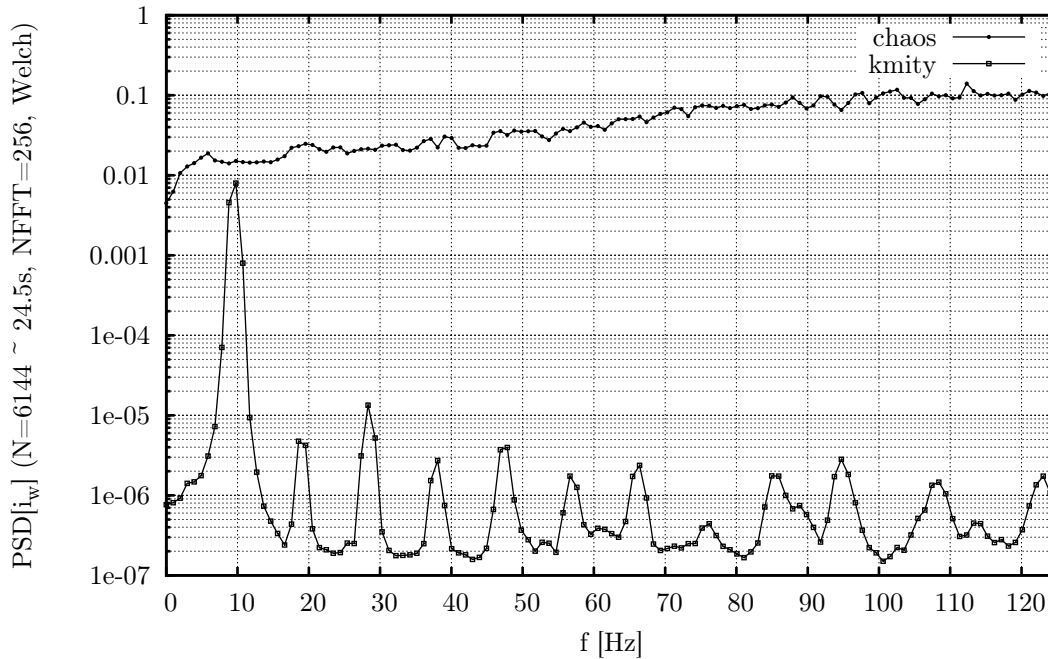


32. časový průběh smyčkou tvořených kmitů



33. časový průběh smyčkou tvořeného chaosu

Vznik kmitů se mi podařilo odůvodnit, princip vzniku chaosu ovšem nikoli. Překvapil mě svojí opakovatelností a setrvalostí. Pravděpodobnost přechodu systému mezi uvedenými třemi stavy se hodně liší s různými činiteli, zejména asi teplotou motoru (a jeho třením), způsobem připevnění jeho setrvačné zátěže k hřídeli apod. Náchylnost ke kmitání i síla kmitů se velmi lišila mezi 12 v robotu použitými servomotory, některé nekmitaly vůbec. Spektrum servomotorem generovaného chaosu oproti spektru pravidelných kmitů je pro zajímavost vykresleno na obr. 34.



34. srovnání výkonové spektrální hustoty kmitů a chaosu

Po podrobném zkoumání jsem došel k závěru, že kmity nejsou způsobeny ani šumem elektroniky, měření polohy, ani lineární nestabilitou či lineárními kmitavými módy v systému. Zjistil jsem, že kmity ustanou, když:

- je polohový PID regulátor ochuzen o integrační složku;
- je kaskáda PID-PI nahrazena pouhým PID (popř. P) regulátorem;
- na hřídel motoru působí zátěžový moment.

Na první pohled by se mohlo zdát, že kmity v řádu 1 LSB A/D převodníku jsou přirozeným důsledkem kvantizace a použití integrační složky v regulátoru, ostatně první bod by tomuto vysvětlení nasvědčoval. Je pravda, že snažit se o regulaci s přesností lepší, než je rozlišení A/D převodníku, je přinejmenším odvážné. Na druhou stranu, měřená poloha je průměrována filtrem délky 80 vzorků a to v některých případech zlepšuje přesnost měření. Měřené napětí z potenciometru ale není téměř vůbec zašuměno, což by paradoxně při zpřesňování měření průměrováním mohlo přinést zlepšení i v ustáleném stavu (noise-dithering). Proti vysvětlení kmitů jako důsledku kvantování měřené polohy a integračního účinku ovšem hovoří skutečnost, že různé motory se v kmitavosti lišily a rovněž poslední bod o vymizení kmitů při zatížení.

Nesmírně důležité je právě to, že kmity zcela vymizí a přesně nulové regulační odchylky (tedy odchylky $|x_w - x| < 1 \text{ LSB}$) je dosaženo v případě, že je hřídel motoru zatížena zvnějšku nějakou silou. Tomu odpovídá i skutečnost, že ke kmitům dochází méně během pohybu motoru a uplatňují se nejvíce právě v ustáleném stavu.

Z tohoto pozorování jsem vyvodil, že rušivé kmity jsou způsobeny *chybným měřením proudu* i a to bohužel již na úrovni elektronického návrhu. V elektronickém návrhu [↔4.3.2.] jsem se dopustil těchto chyb v měření proudu, z nichž pravděpodobně alespoň jedna působí předmětné rušivé kmity:

- Nelinearita způsobená nespojitostí kolem nuly – závislost mezi měřeným proudem a údajem A/D převodníku by měla být až na kvantizaci přímkou, procházející počátkem. Ve skutečnosti ovšem nemusí procházet počátkem úplně přesně a taková nespojitost

zrovna v citlivém bodě počátku může způsobit v smyčce kmitání. Pomocí odpovídající nelinearity se mi podařilo podobné kmity nasimulovat. Naproti tomu pokus o narovnaní charakteristiky jejím stejnosměrným posunutím doprostřed rozsahu A/D převodníku a odečítáním klidové hodnoty ztroskotat, naopak, kmity poněkud vzrostly.

- Unipolární měření, nahrazující bipolární – jak již bylo podotknuto v [↔4.3.3.]. Závažnost této chyby je ovšem oslabena skutečností, že při zatížení hřídele vnější silou (a tedy vyvoláním nenulového klidového proudu vinutím) kmity mizejí.
- Proud H-můstkem se liší od proudu motorem – též viz [↔4.3.3.].

Je nutné si uvědomit závažnost kmitů zejména pro úlohu krácejícího robota. I zdánlivě zanedbatelný rozkmit ± 1 LSB tj. $0,4^\circ$, činí při délce natažené nohy 240 mm na druhé straně rozkmit ~ 2 mm. S tím je samozřejmě spojeno kmitání značných setrvačných hmot a z toho plynoucích poruch, přenášených do ostatních servomotorů. Výsledkem je, že robot se při chůzi značně třese, ovšem kmity zůstávají ve svých mezích a jsou tak k požadované trajektorii vlastně přičítány.

6.3.4. Zhodnocení

Řízení robota proudově omezeným kaskádním regulátorem již je bezpečné, elektronika i vinutí motorů jsou chráněny proti nadproudovému poškození. Integrační složka napomáhá držet správnou trajektorii bez systematické chyby. V součinnosti s ní vydatně prospělo přepracování mechanického uložení svislé osy kyčelního kloubu, které umožnilo tvorbu opakovatelné trajektorie a myslitelného měření skutečné polohy (úhlů) robota. Stabilita robota s jediným regulátorem ve všech kloubech a všech okamžicích při použití integrační složky a se zpožděním ve vnořené proudové smyčce již není tak samozřejmá, jako při řízení prostým P regulátorem. Nastavení PID regulátoru je třeba provést opatrně a i při takovém se zdálo, že robot je v některých pozicích blížký nestabilitě.

Jak by bylo možné zlepšit jakost regulace? V SISO přístupu by bylo vhodné nastavit každý regulátor dle konkrétní setrvačné zátěže, což by ovšem zejména v případě některých kloubů (kotníků) bylo vhodné učinit spolu s přepínáním regulátoru (modelu) podle fáze chůze – stoje na jedné, či obou nohou. Přínosem by mohlo být použití prediktoru dopravního zpoždění (3 kroky), obecně regulátoru, kraticím póly z^{-1} . Nejsmělejší cílem by bylo použití MIMO regulátoru. Úloha identifikace robota jako celku je ale obtížná. I při ní by bylo nutné rozdělit dynamiku robota podle fází chůze a přepínat model.

Vadou výsledného kaskádního řízení s omezením proudu je kmitání, způsobené chybně navrženou elektronikou. Tyto vady by měly být odstraněny po přepracování měření proudu dle [↔4.3.3.].

7. Inverzní kinemacká úloha

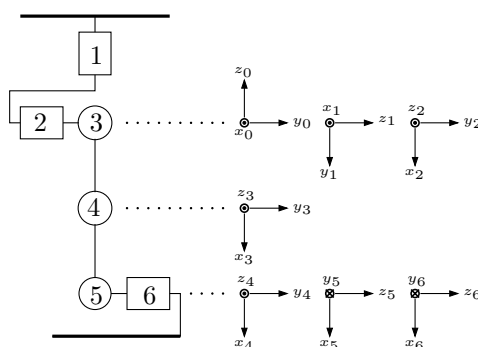
Z hlediska kinemackého je každá z nohou robota řetězcem, spojeným otočnými klouby o celkem 6 osách pohybu. Řešme inverzní kinemackou úlohu jedné nohy pro případ, že robot není v pohybu nijak omezen, tj. např. nestojí oběma nohama na zemi. V takovém případě je noha regulárním sériovým manipulátorem typu 6R [17].

Definujme vstup úlohy jako polohu a orientaci chodidla nohy vůči počátku spojenému s pánví. Poloha je dána maticí homogenní transformace \mathbf{T} .

V následujících matematických výrazech byly použity zkratky: c_i pro $\cos \vartheta_i$, s_i pro $\sin \vartheta_i$, c_{i+j} , s_{i+j} pro $\cos(\vartheta_i + \vartheta_j)$, $\sin(\vartheta_i + \vartheta_j)$. $(\mathbf{M})_{s\dots u, r\dots t}$ pro podmatici \mathbf{M} tvořenou sloupci $s\dots u$ a řádky $r\dots t$. Funkce $\varphi = \arctg x$ vrací nejednoznačnou hodnotu $\varphi : \tg \varphi = \tg(\pi + \varphi) = x$, zatímco $\varphi = \arctg(y, x)$ vrací jednoznačnou hodnotu úhlu $\varphi : \cos \varphi = x/\sqrt{x^2 + y^2}$, $\sin \varphi = y/\sqrt{x^2 + y^2}$ (odpovídá knihovní funkci programovacího jazyka C jménem `atan2`).

7.1. Popis nohy robota dle Denavit-Hartenbergovy konvence

Osy 1, 2, 3 tvoří kulový kloub kyčle, do něhož rovněž umístíme počátek souřadného systému pánve. Jednoosý kolenní kloub se otáčí v ose 4. Osy 5, 6 tvoří dvouosý kotníkový kloub, do něhož souřadný systém chodidla, daný vůči pánvi transformací \mathbf{T} . Pro další použití je pak možno pánevní i chodidlový souřadný systém posunout konstantní transformací tak, aby lépe odpovídal praktickým požadavkům – pánevní systém umístit doprostřed pánve (pro obě nohy stejný) a chodidlový systém na dolní plochu chodidla.



35. souřadné systémy vzhledem k osám nožních kloubů

Volbu souřadných systémů pro jednotlivá tělesa ukazuje obr. 35. Vzdálenost mezi kyčlí a kolenem (délka stehna) a kolenem a kotníkem (délka lýtky) je shodná, rovna L . Popis nohy v D-H¹⁴ notaci je dán těmito parametry:

¹⁴ Denavit-Hartenbergova notace pro popis otáčení a posouvání v prostoru, používající homogenní souřadnice a zobrazení maticemi 4×4 ; [17]

i	a_i	d_i	α_i	ϑ_i
1	0	0	$-\frac{\pi}{2}$	ϑ_1
2	0	0	$\frac{\pi}{2}$	ϑ_2
3	L	0	0	ϑ_3
4	L	0	0	ϑ_4
5	0	0	$-\frac{\pi}{2}$	ϑ_5
6	0	0	0	ϑ_6

Přímou transformací otáčením v i -tém kloubu značme \mathbf{A}_{i+1}^i , zpětnou transformaci $\mathbf{A}_i^{i+1} = (\mathbf{A}_{i+1}^i)^{-1}$. Pro více následujících transformací píšme zkráceně $\mathbf{A}_j^i = \mathbf{A}_{i+1}^i \mathbf{A}_{i+2}^{i+1} \dots \mathbf{A}_j^{i+n}$, $\mathbf{A}_i^j = (\mathbf{A}_j^i)^{-1}$. Všechny klouby jsou otočné, $\mathbf{A}_{i+1}^i = \mathbf{A}_{i+1}^i(\vartheta_i)$.

7.2. Odvození řešení inverzní kinematické úlohy

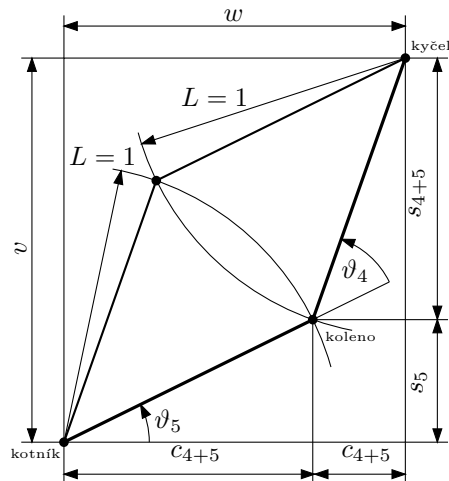
Pro sériový manipulátor se šesti stupni volnosti, obsahujícím tříosý kloub, je inverzní kinematická úloha řešitelná analyticky ([17], 2.3.3.).

Rovnici sériového manipulátoru $\mathbf{T} = \mathbf{A}_1^0 \mathbf{A}_2^1 \mathbf{A}_3^2 \mathbf{A}_4^3 \mathbf{A}_5^4 \mathbf{A}_6^5$ upravíme do podoby $\mathbf{T}^{-1} \mathbf{A}_1^0 \mathbf{A}_2^1 = \mathbf{A}_5^6 \mathbf{A}_4^5 \mathbf{A}_3^4 \mathbf{A}_2^3$, kde souřadnice transformací na obou stranách rovnice vyjadřují polohu kyčelního kloubu vůči chodidlu (6. souřadnému systému). Transformace $\mathbf{A}_1^0 \mathbf{A}_2^1$ polohu nemění, pouze otáčejí souřadný systém $(\mathbf{A}_1^0 \mathbf{A}_2^1)_{1\dots 3,4} = (0, 0, 0)^T$, proto souřadnice kyčle vůči chodidlu $(x, y, z)^T = (\mathbf{T}^{-1} \mathbf{A}_1^0 \mathbf{A}_2^1)_{1\dots 3,4} = (\mathbf{T}^{-1})_{1\dots 3,4}$. Vyjádřením těchto souřadnic z pravé strany dostaneme:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = (\mathbf{A}_5^6 \mathbf{A}_4^5 \mathbf{A}_3^4 \mathbf{A}_2^3)_{1\dots 3,4} = L \begin{pmatrix} -c_6(c_{4+5} + c_5) \\ -s_6(c_{4+5} + c_5) \\ s_{4+5} + s_5 \end{pmatrix}.$$

Nyní můžeme vypočítat první z hledaných úhlů, $\vartheta_6 = \arctg y/x$. Zde vzniká nejednoznačnost (1), neboť rovnici vyhoví v rozsahu 2π dva úhly o π posunutě.

Po dosazení $w = -\sqrt{x^2 + y^2}/L$, $v = z/L$ získáme rovnici $(w, v)^T = (c_{4+5} + c_5, s_{4+5} + s_5)^T$. Rovnice popisuje průnik dvou jednotkových kružnic, lýtkové se středem v kotníku a stehenní se středem v kyčli, v rovině kolmé na osy 4, 5 (obr. 36).



36. řešení úhlů v rovině kotník-koleno-kyčel

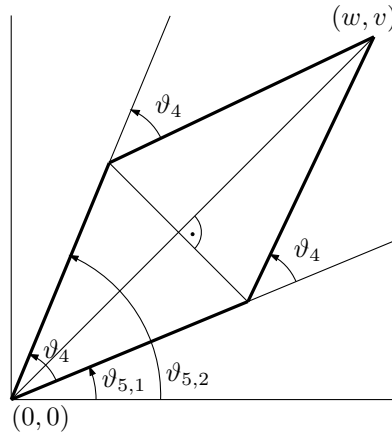
Rovnice průniku kružnic vede na kvadratickou rovnici, jejímž řešením je

$$c_{5_{1,2}} = \frac{1}{2} \left(w \pm v \sqrt{\frac{4}{w^2 + v^2} - 1} \right),$$

$$s_{5_{1,2}} = \frac{1}{2} \frac{w^2 + v^2 - 2wc_{5_{1,2}}}{v}.$$

Rovnice může mít v reálném případě dvě řešení, odpovídající “levé” a “pravé” konfiguraci manipulátoru, což přináší další nejednoznačnost (2). Ve výjimečném případě je řešení jedno, odpovídající rovně natažené noze. Pokud rovnice nemá řešení, znamená to, že cíl je nedosažitelný pro danou délku stehne a lýtka – je vzdálen od kyčle více než $2L$.

Diskriminant je nedefinován v případě, že cílový bod (kotník) je totožný s počátečním (kyčel), $w^2 + v^2 = 0$. V tomto (pro praxi nezajímavém) případě má úloha nekonečně mnoho řešení a c_5 můžeme volit libovolně. Uvedená rovnice pro s_5 je ovšem nevýhodná vinou dělení v . Pro $v \rightarrow 0$ ztrácí výpočet v plovoucí řádové čárce přesnost. Proto nahradíme řešení kvadratické rovnice jiným výpočtem, založeným na geometrických faktech $|\vartheta_4| = |\vartheta_{5_2} - \vartheta_{5_1}|$ a $2 \cos \vartheta_4/2 = \sqrt{w^2 + v^2}$ (viz obr. 37).



37. dvě možná řešení úhlů roviny kotník-koleno-kyčel

Potom jsou hledané úhly ϑ_5 , ϑ_4 :

$$\vartheta_{4_{1,2}} = \pm 2 \arccos \frac{\sqrt{w^2 + v^2}}{2},$$

$$\vartheta_{5_{1,2}} = \arctg(v, w) - \vartheta_{4_{1,2}}.$$

Funkce $\arctg(y, x)$ musí být dodefinována pro případ $x = 0 \wedge y = 0$ (obdoba nedefinovaného diskriminantu), případ nedefinovaného $\arccos(\cdot)$ znamená cíl mimo dosah. V případě cíle mimo dosah je možno oříznout argument $\arccos(\cdot)$ na ± 1 , aby se robot cíli alespoň nanejvýš přiblížil.

Zbývá určit úhly $\vartheta_{1...3}$, tvořící prostorové otočení kyčelního kulového kloubu. Známe již úhly $\vartheta_{4...6}$ a jimi danou transformaci $\mathbf{A}_6^3 = \mathbf{A}_4^3 \mathbf{A}_5^4 \mathbf{A}_6^5$. Z rovnice manipulátoru můžeme psát

$$\mathbf{T}\mathbf{A}_3^6(\vartheta_4, \vartheta_5, \vartheta_6) = \mathbf{A}_1^0(\vartheta_1) \mathbf{A}_2^1(\vartheta_2) \mathbf{A}_3^2(\vartheta_3),$$

z čehož pro samotné rotační části matic platí

$$\mathbf{R}_3^0 = \mathbf{T}_{1...3,1...3} \mathbf{R}_3^6(\vartheta_4, \vartheta_5, \vartheta_6) = \mathbf{R}_1^0(\vartheta_1) \mathbf{R}_2^1(\vartheta_2) \mathbf{R}_3^2(\vartheta_3),$$

kde $\mathbf{R}_j^i = (\mathbf{A}_j^i)_{1\dots 3, 1\dots 3}$. Úloha nalezení úhlů $\vartheta_{1\dots 3}$ při známé matici \mathbf{R}_3^0 je obdobou určení Eulerových úhlů dané rotace. Matici \mathbf{R}_3^0 vypočteme přímým násobením:

$$\mathbf{R}_3^0 = \mathbf{T}_{1\dots 3, 1\dots 3} \begin{pmatrix} c_6 c_{4+5} & c_6 s_{4+5} & -s_6 \\ -s_6 c_{4+5} & -s_6 s_{4+5} & -c_6 \\ -s_{4+5} & c_{4+5} & 0 \end{pmatrix} = (r_{ij}).$$

Úhly určíme z upravené rovnice

$$\begin{aligned} \mathbf{R}_3^0 &= \mathbf{R}_1^0 \mathbf{R}_2^1 \mathbf{R}_3^2, \\ \mathbf{R}_3^0 \mathbf{R}_2^3 &= \mathbf{R}_1^0 \mathbf{R}_2^1, \\ \begin{pmatrix} c_3 r_{11} - s_3 r_{12}, & s_3 r_{11} + c_3 r_{12}, & r_{13} \\ c_3 r_{21} - s_3 r_{22}, & s_3 r_{21} + c_3 r_{22}, & r_{23} \\ c_3 r_{31} - s_3 r_{32}, & s_3 r_{31} + c_3 r_{32}, & r_{33} \end{pmatrix} &= \begin{pmatrix} c_1 c_2 & -s_1 & c_1 s_2 \\ s_1 c_2 & c_1 & -s_1 c_2 \\ -s_2 & 0 & c_2 \end{pmatrix}. \end{aligned}$$

Nejprve $\vartheta_3 = \arctg\left(-\frac{r_{32}}{r_{31}}\right)$, což přináší další nejednoznačnost (3) o π posunutých Eulerových úhlů. Nakonec vypočteme $\vartheta_2 = \arctg(s_3 r_{32} - c_3 r_{31}, r_{33})$, $\vartheta_1 = \arctg(-s_3 r_{11} - c_3 r_{12}, s_3 r_{21} + c_3 r_{22})$.

Nejednoznačnosti (1), (2), (3) tvoří 8 různých řešení pro každou zadanou transformaci, vyjma výjimečného případu přímo natažené nohy (2), kdy jsou řešení jen 4.

7.3. Poznámka k výpočtu různých řešení

Mezi vypočtenými úhly jednotlivých 8 řešení jsou poměrně jednoduché závislosti, popsatelné součty a rozdíly. Stačí tedy výše uvedeným způsobem vypočíst pouze jedno řešení a zbylých 7 dopočítat kombinacemi součtů a rozdílů.

Následující vztahy uvedu bez důkazu – došel jsem k nim empiricky. Pro úhly 4, 5 je závislost na nejednoznačnosti (1) zřejmá z náčrtku zrcadlení i z rovnic, pro Eulerovy úhly 1, 2 se mi ale rozumné odvození ani přehledný náčrtek sestavit nepodařilo.

Tabulka ukazuje, jak se z jednoho vypočteného řešení $\vartheta_{1\dots 6}$ určí řešení $\vartheta_{1\dots 6}^*$ příslušné nějaké kombinaci nejednoznačností (1), (2), (3):

(1)	0	0	0	0	1	1	1	1
(2)	0	0	1	1	0	0	1	1
(3)	0	1	0	1	0	1	0	1
ϑ_6^*	ϑ_6				$\pi + \vartheta_6$			
ϑ_5^*	ϑ_{5_1}		ϑ_{5_2}		$\pi - \vartheta_{5_1}$		$\pi - \vartheta_{5_2}$	
ϑ_4^*	ϑ_4		$-\vartheta_4$		$-\vartheta_4$		ϑ_4	
ϑ_3^*	ϑ_3	$\pi + \vartheta_3$	$\vartheta_3 + \vartheta_4$	$\pi + \vartheta_3 + \vartheta_4$	$\pi - \vartheta_3$	$-\vartheta_3$	$\pi - \vartheta_3 - \vartheta_4$	$-\vartheta_3 - \vartheta_4$
ϑ_2^*	ϑ_2	$-\vartheta_2$	ϑ_2	$-\vartheta_2$	$\vartheta_2 - \pi$	$\pi - \vartheta_2$	$\vartheta_2 - \pi$	$\pi - \vartheta_2$
ϑ_1^*	ϑ_1	$\pi + \vartheta_1$	ϑ_1	$\pi + \vartheta_1$	ϑ_1	$\pi + \vartheta_1$	ϑ_1	$\pi + \vartheta_1$

Jednotlivé osy mají 2, 4 nebo 8 různých řešení. Ta je možno uložit do tabulek a odkázat se na ně indexem pro danou kombinaci nejednoznačností, daným následující tabulkou:

ϑ	počet různých hodnot	index
6	2	$i_{(1)}$
5	4	$2i_{(1)} + i_{(2)}$
4	2	$i_{(1)} \oplus i_{(2)}$
3	8	$4i_{(1)} + 2i_{(2)} + i_{(3)}$
2	4	$2i_{(1)} + i_{(3)}$
1	2	$i_{(3)}$

7.4. Zhodnocení

Inverzní kinematická úloha jedné nohy kráčejícího robota, řešící nastavení úhlů 6 os kloubů nohy pro danou polohu a orientaci chodidla vůči pánvi, byla vyřešena analyticky s použitím pěti cyklometrických funkcí, jednoho maticového násobení a násobení matice a vektoru dimenze 3. Algoritmus je numericky stabilní pro libovolný vstup a dává všechna řešení pro neomezené otáčení o 360° všech osách.

8. Závěr

8.1. Výsledky

- V rámci práce byla vyrobena mechanická konstrukce robota, řídicí i výkonná elektronika, ovládací a animační software.
- Chůze v 12 stupních volnosti bylo dosaženo již s velmi jednoduchými prostředky: P regulátor, pevná, ručně animovaná trajektorie.
- Bylo zjištěno, že pro účelné zpětnovazební řízení je nezbytná kvalitní mechanická konstrukce bez hysterezí – vzhledem k tomu, že ty se v kloubech vyskytovaly až za měřenou veličinou (polohou), tj. mimo zpětnovazební smyčku.
- S použitými servomotory a celkovou hmotností robota je nezbytné používat omezení proudu, jinak docházelo neustále k přehřívání a ničení součástí. Pro tento účel byl navržen z hlediska lineárního řízení vhodný proudový regulátor s omezením.
- Byly odvozeny vztahy, užitečné při návrhu vnořených zpětnovazebních smyček se změnou vzorkovací frekvence.
- Analytické řešení inverzní kinematické úlohy bylo umožněno koncepcí mechanické konstrukce. Byl nalezen rychlý a numericky stabilní algoritmus výpočtu všech jejích řešení.

8.2. Náměty na pokračování

- odstranění vady měření proudu, jejíž vinou servomotory mírně kmitají;
- určení hmotností a setrvačností těles robota pro statickém i dynamickém modelování;
- samočinná tvorba statické trajektorie na základě statického modelu;
- návrh MIMO regulátoru na základě dynamického modelu;
- přídatná čidla (tíže, tlak);
- řízení s pružnou trajektorií.

9. Literatura

- [1] C. Chevallereau, E. R. Westervelt, J. W. Grizzle: “Asymptotic stabilization of a five-link, four-actuator, planar bipedal runner” *43th IEEE Conference on Decision and Control*, 2004.
- [2] H. Choset, K. M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. E. Kavraki, S. Thrun: *Principles of robot motion: theory, algorithms, and implementation*, MIT Press, 2005.
- [3] Boston Dynamics, Inc.: “The “BigDog” robot”.
<http://www.bdi.com/content/sec.php?section=BigDog>
- [4] M. Albero, F. Blanes, G. Benet, P. Perez, J. E. Simo, J. Coronel: “Distributed real time architecture for small biped robot YABiRo” *IEEE Intl. Symposium on Computational Intelligence in Robotics and Automation*, Departamento de Informatica de Sistemas y Computadores, Univ. Politecnica de Valencia, Spain, 2005.
- [5] C. Ridderström, J. Ingvast, F. Hardarson, M. Gudmundsson, M. Hellgren, J. Wikander, T. Wadden, H. Reh binder: “The basic design on the quadruped robot Warp1” *Intl. Conference on Climbing and Walking Robots*, Center of Autonomous Systems, KTH, 2000.
- [6] F. Yamasaki, T. Miyashita, T. Matsui, H. Kitano: “PINO the Humanoid: A basic architecture” *4th Intl. Workshop on RoboCup*, 2000.
- [7] *CAN Specification 2.0B*, Robert Bosch GmbH, 1991.
<http://www.semiconductors.bosch.de/pdf/can2spec.pdf>
- [8] P. Píša, F. Vacek: “Open source components for the CAN bus” *Proceedings of the 5th Real-Time Linux Workshop*, 2003.
- [9] J. Zikmund, S. Čelikovský, C. H. Moog: “Nonlinear Control Design for the Acrobat” *3rd IFAC Symposium on System, Structure and Control (SSSC'07)*, 2007.
- [10] L. B. Freidovich, H. K. Khalil: “Robust Feedback Linearization using Extended High-Gain Observers” *45th IEEE Conference on Decision and Control*, 2006.
- [11] K. J. Åström, B. Wittenmark: *Computer-Controlled Systems – Theory and Design*, Prentice Hall, 1997.
- [12] J. John: *Systémy a řízení*, Vydavatelství ČVUT, 1996.
- [13] Z. Song, D. Sarwate: “The frequency spectrum of pulsewidth modulated signals”, 2002.
- [14] B. D. O. Anderson: “Controller design: moving from theory to practice” *IEEE Control Systems Magazine*, 1993.
- [15] L. Ljung: *System Identification – Theory For the User, 2nd edition*, PTR Prentice Hall, 1999.
- [16] K. J. Åström, H. Hagglund: *PID controllers: Theory, design, and tuning, 2nd ed.*, Instrument Society of America, 1995.
- [17] H. Asada, J.-J. E. Slotine: *Robot Analysis and Control*, John Wiley and Sons, 1992.

10. Příloha – schémata a osazení

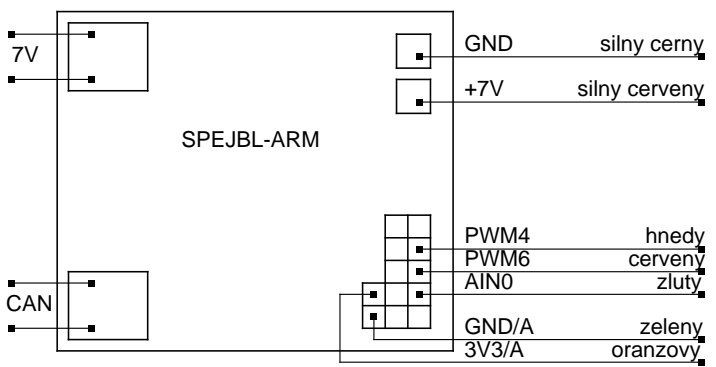
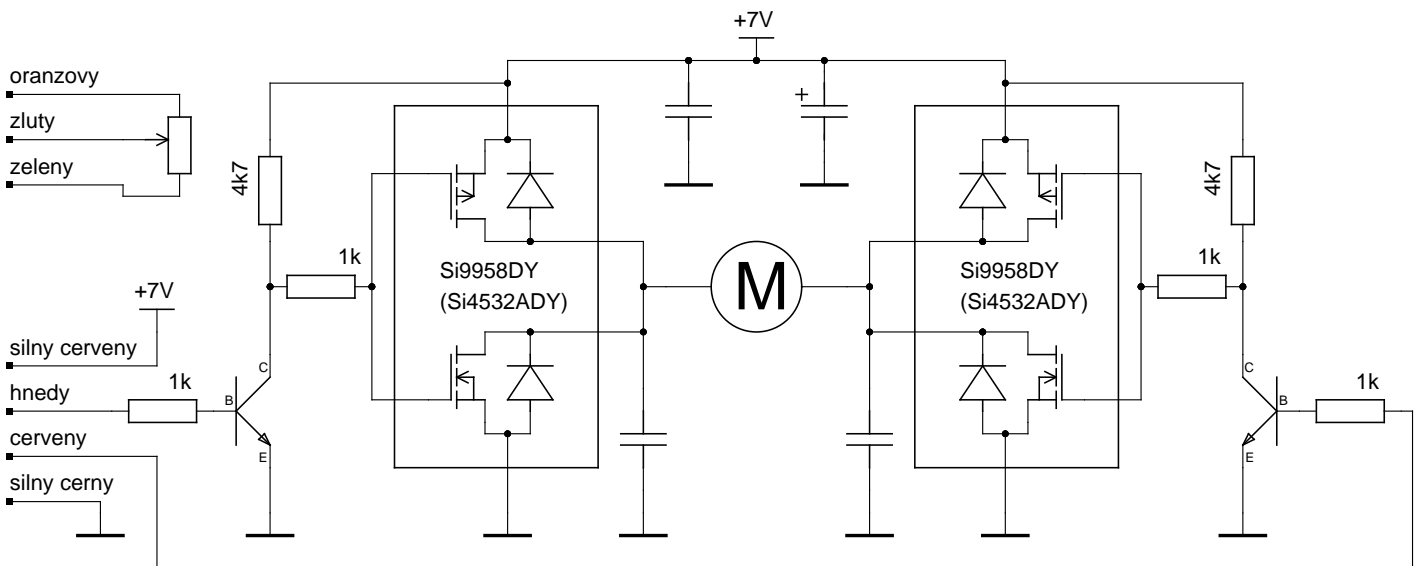
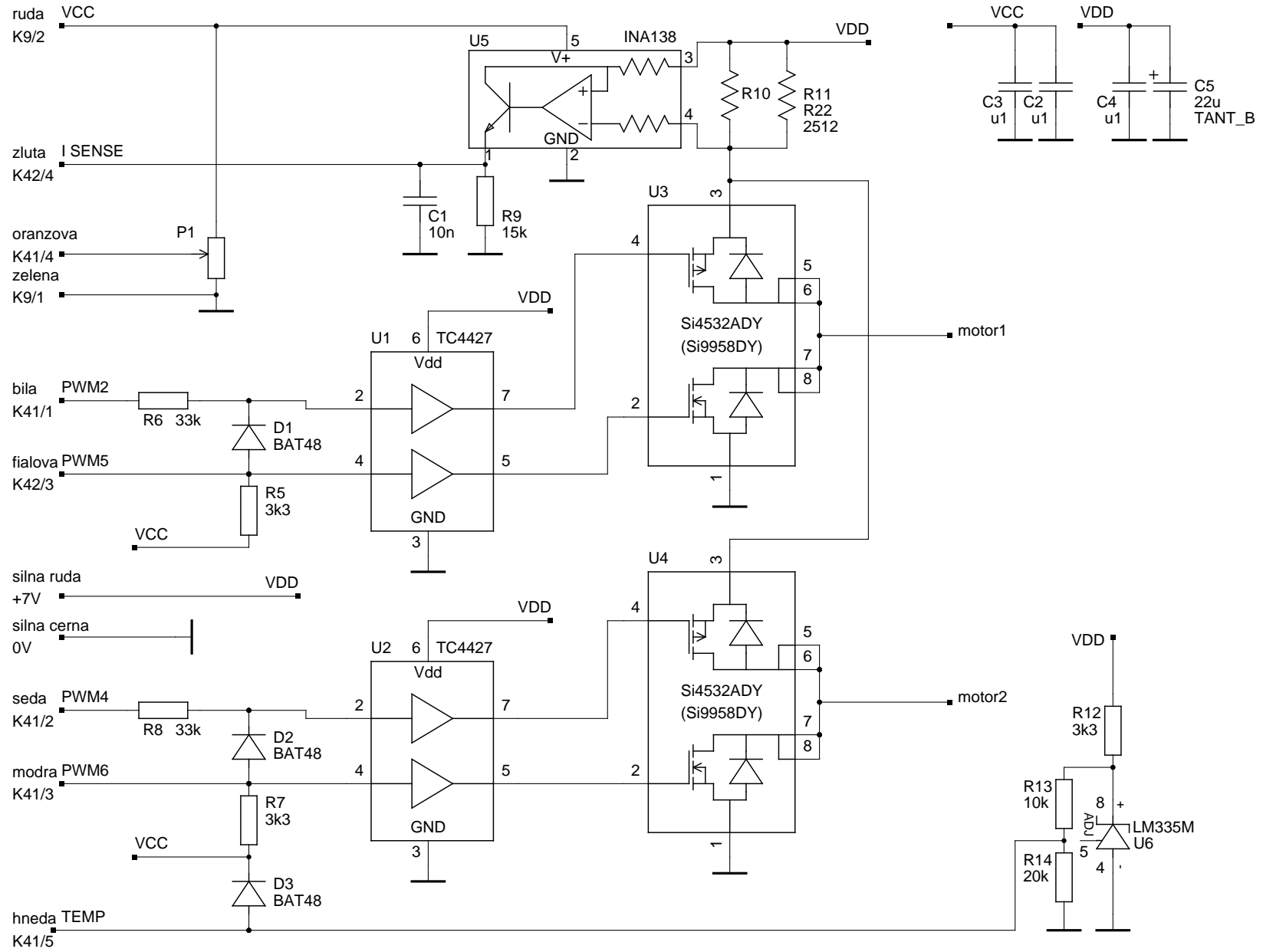
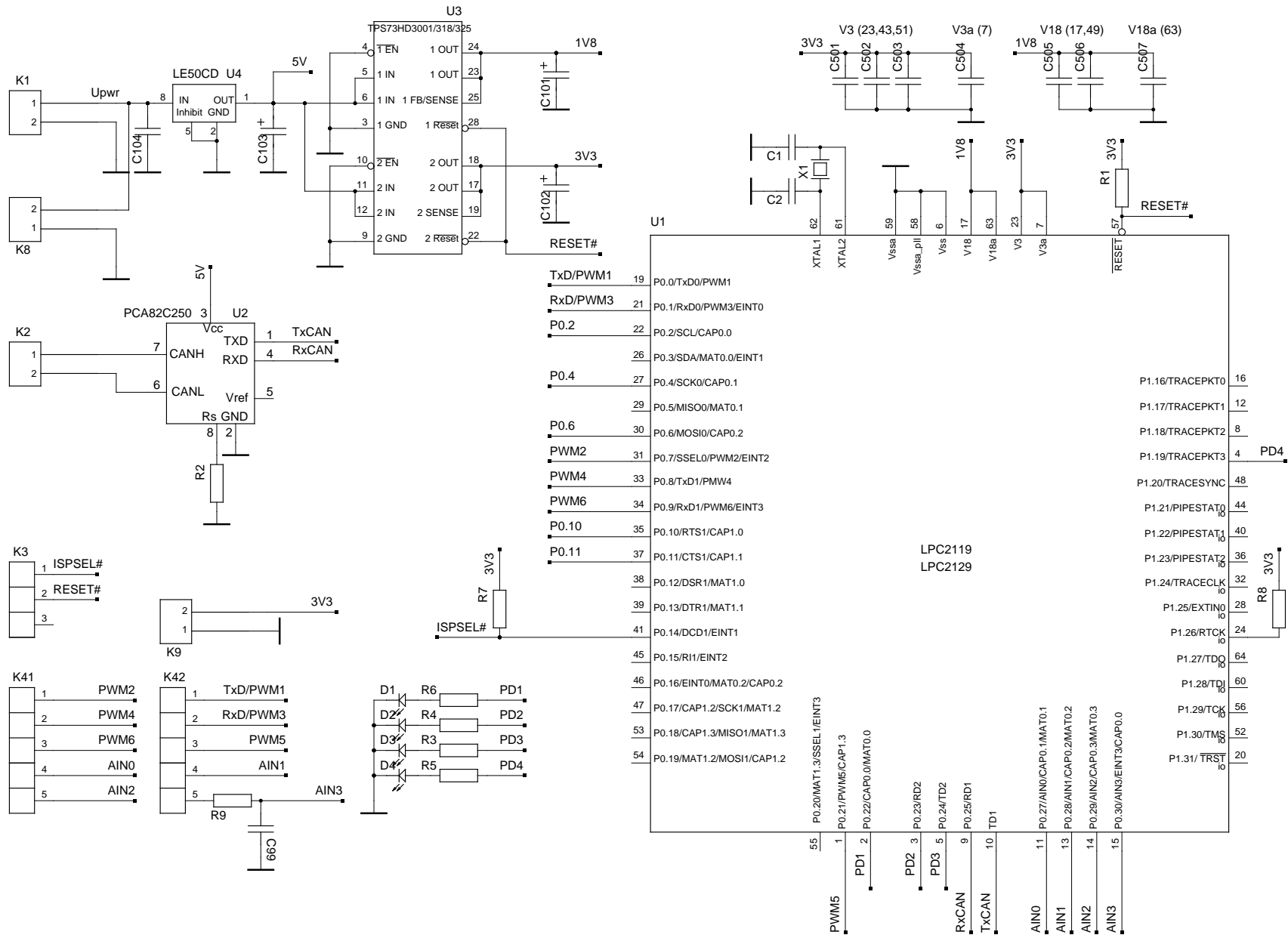
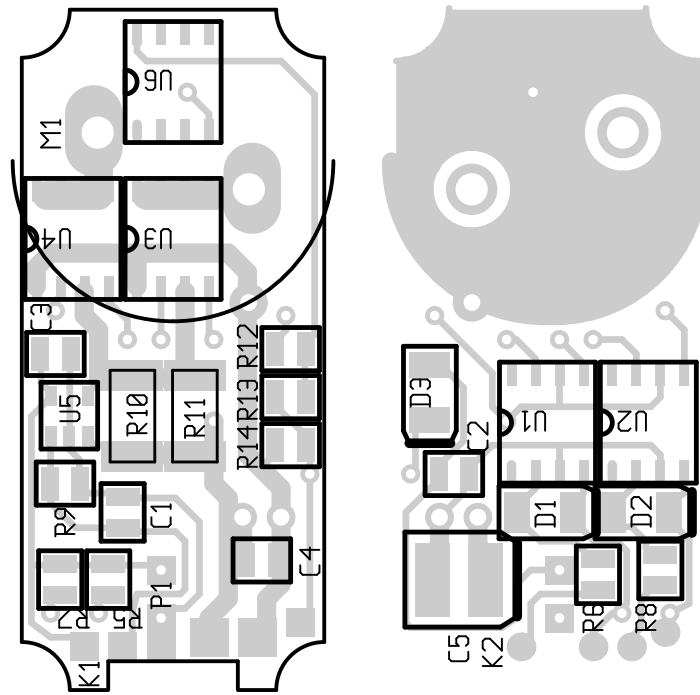


schéma původní elektroniky servomotoru

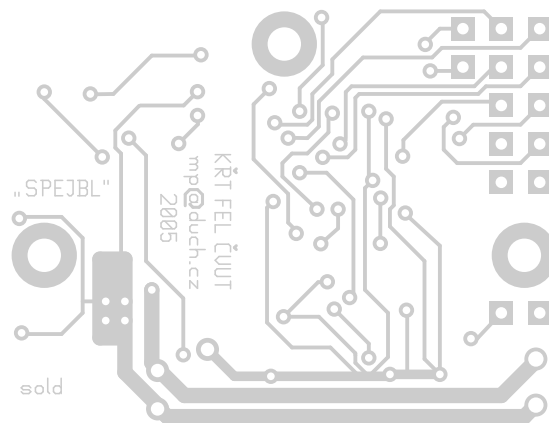
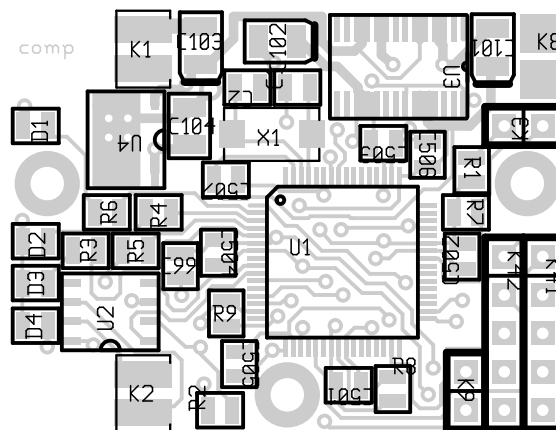
schéma nové elektroniky servomotoru







osazovací plán desky elektroniky servomotoru

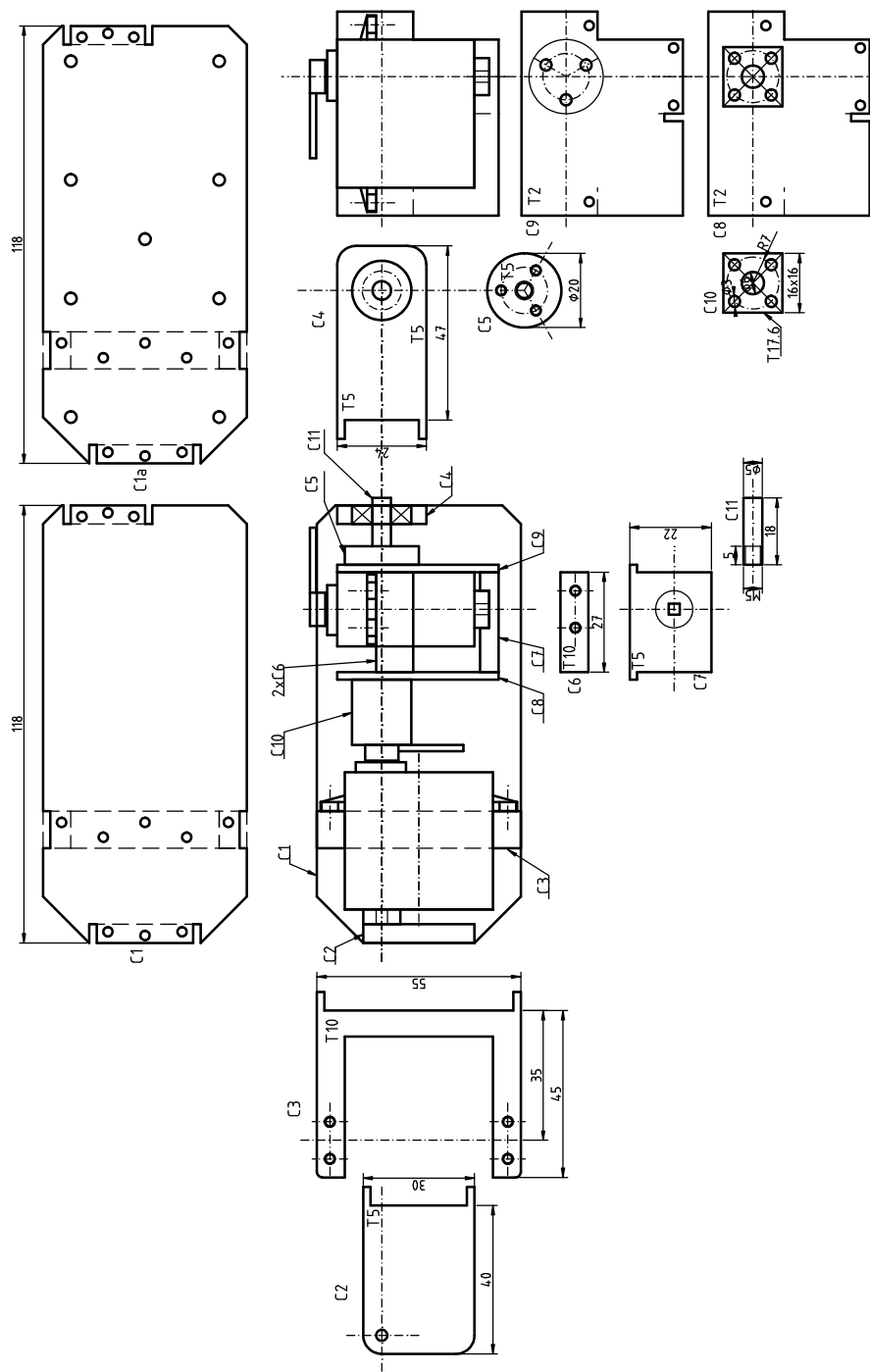


osazovací plán desky SPEJBL-ARM

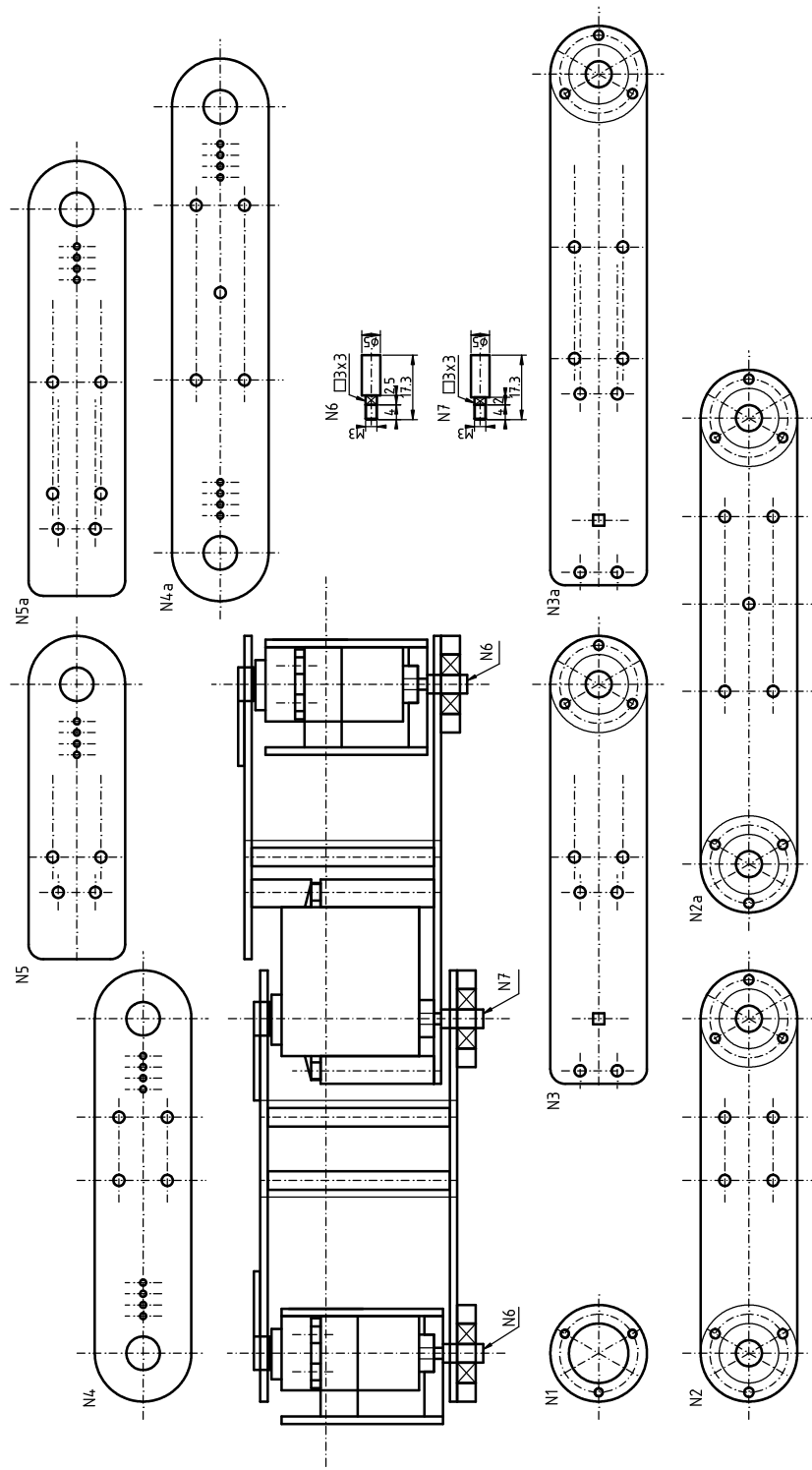
Seznam součástek desky SPEJBL-ARM

označení	typ / hodnota	pouzdro	poznámka
C1, C2	56p	0805	keramické
C99			neosazen (dle potřeby)
C101, C102, C103	4u7	velikost A	tantalové
C104	2u2 (1u)	1206	keramický
C501, ..., C507	u1	0805	keramické
D1, D2, D3, D4	LED (R, Y, G, B)	0805	
K1, K2	PS 25/2G		(GES)
K3, K41, K42, K8, K9			pájecí plošky
R1, R7, R8	10k	0805	
R2	0j	0805	
R3, R4, R5, R6	k47	0805	
R9			neosazen (dle potřeby)
U1	LPC2119FBD64	LQFP64	Philips (Spoerle)
U2	PCA82C250T	SO8	
U3	TPS73HD318	TSSOP28	TI (Spoerle)
U4	LE50CD	SO8	
X1	10MHz	HCX-6FA	Hosonic (Spezial)

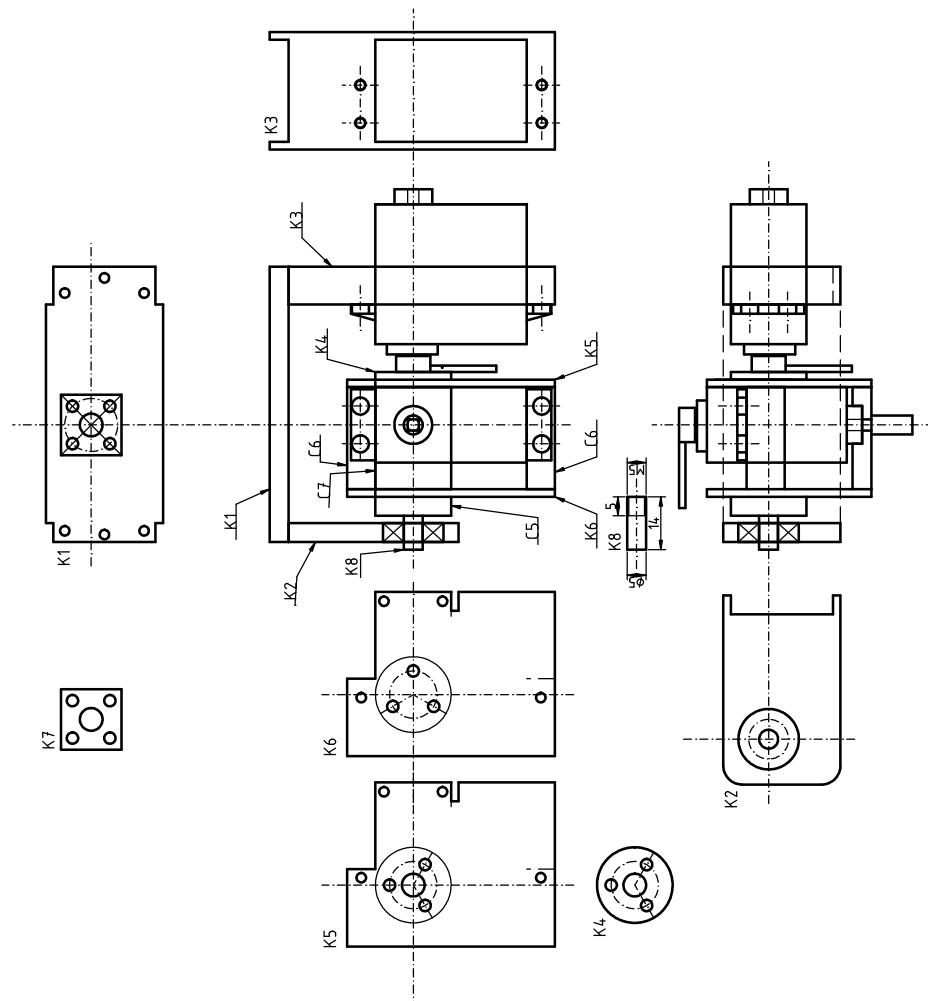
11. Příloha – výkresy mechanických dílů



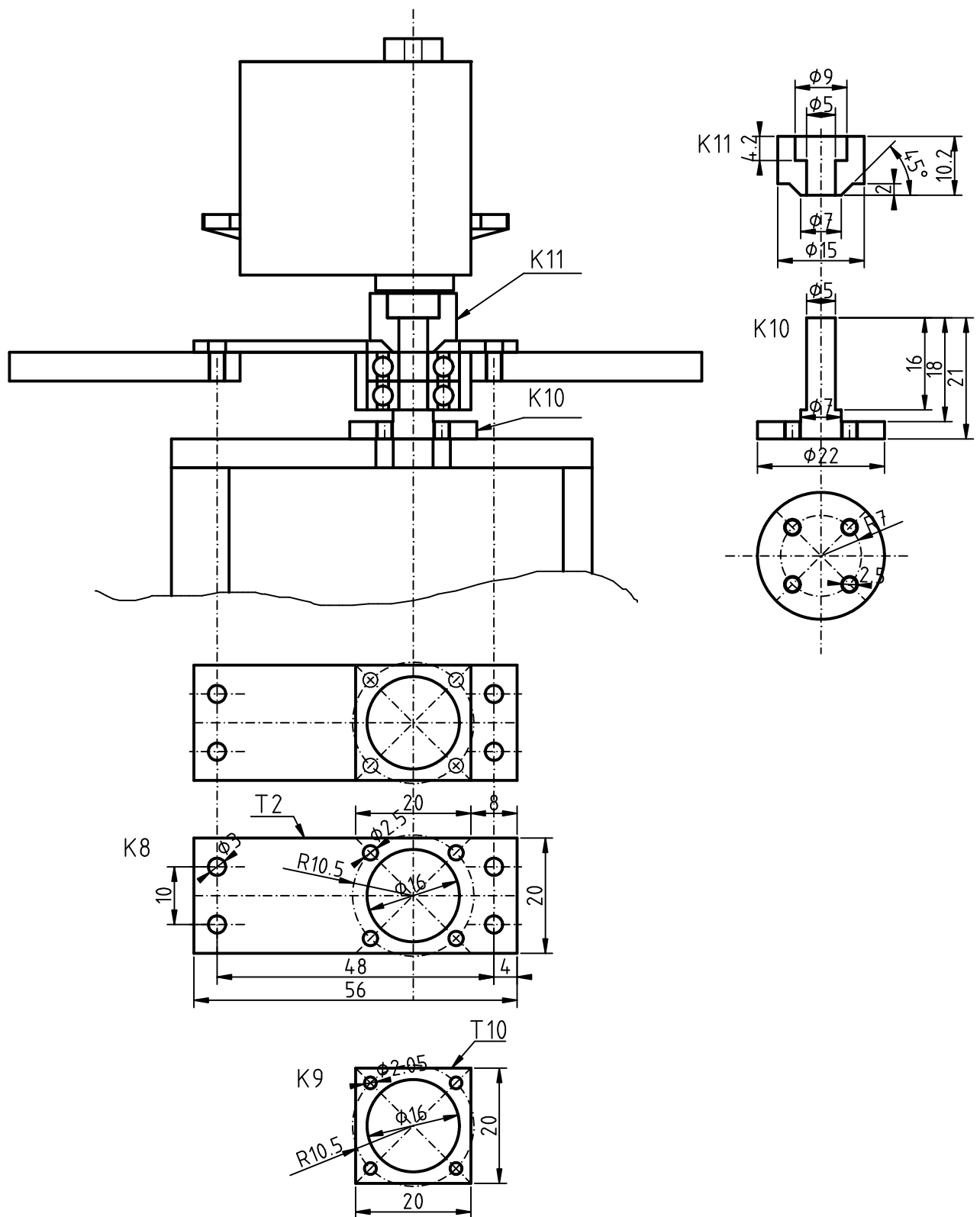
chodidlo



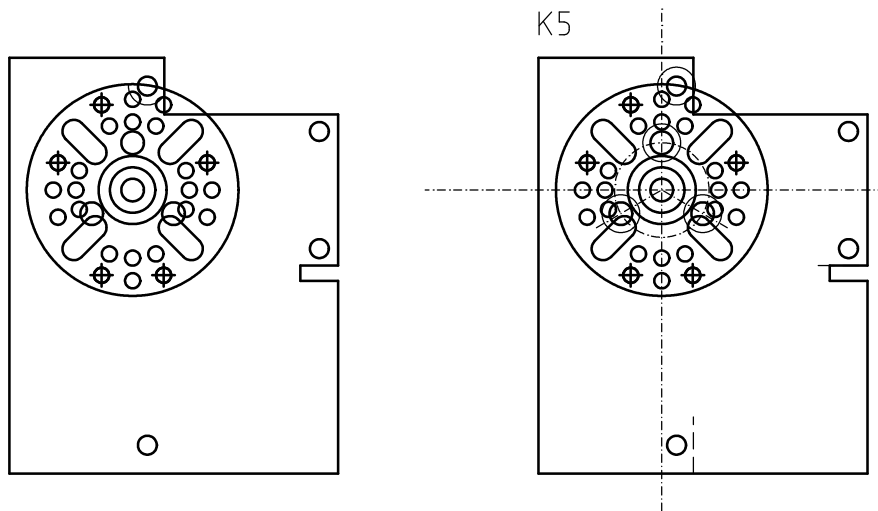
noha



kyčel, spodní část

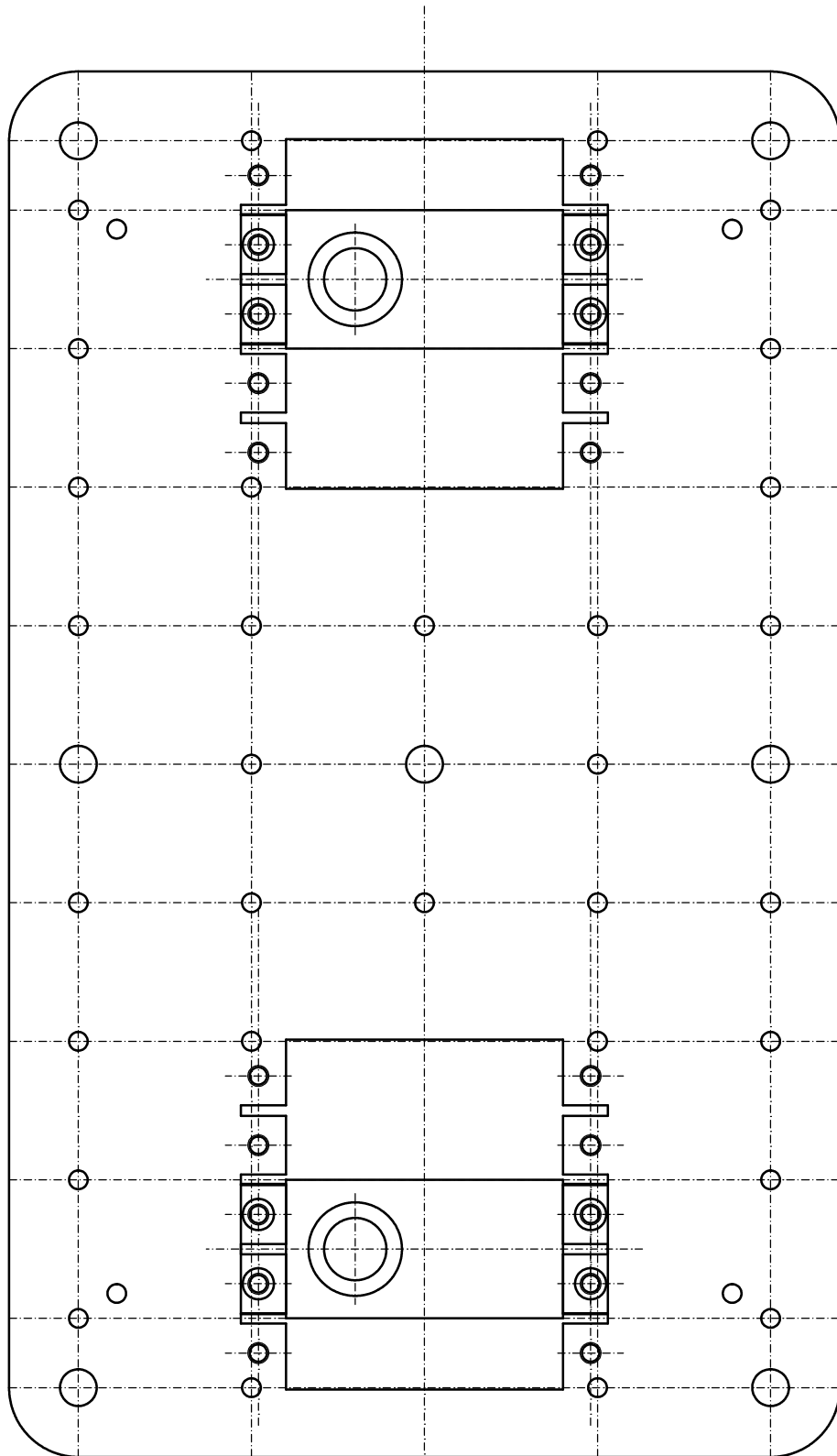


kyčel, horní část, nové uložení

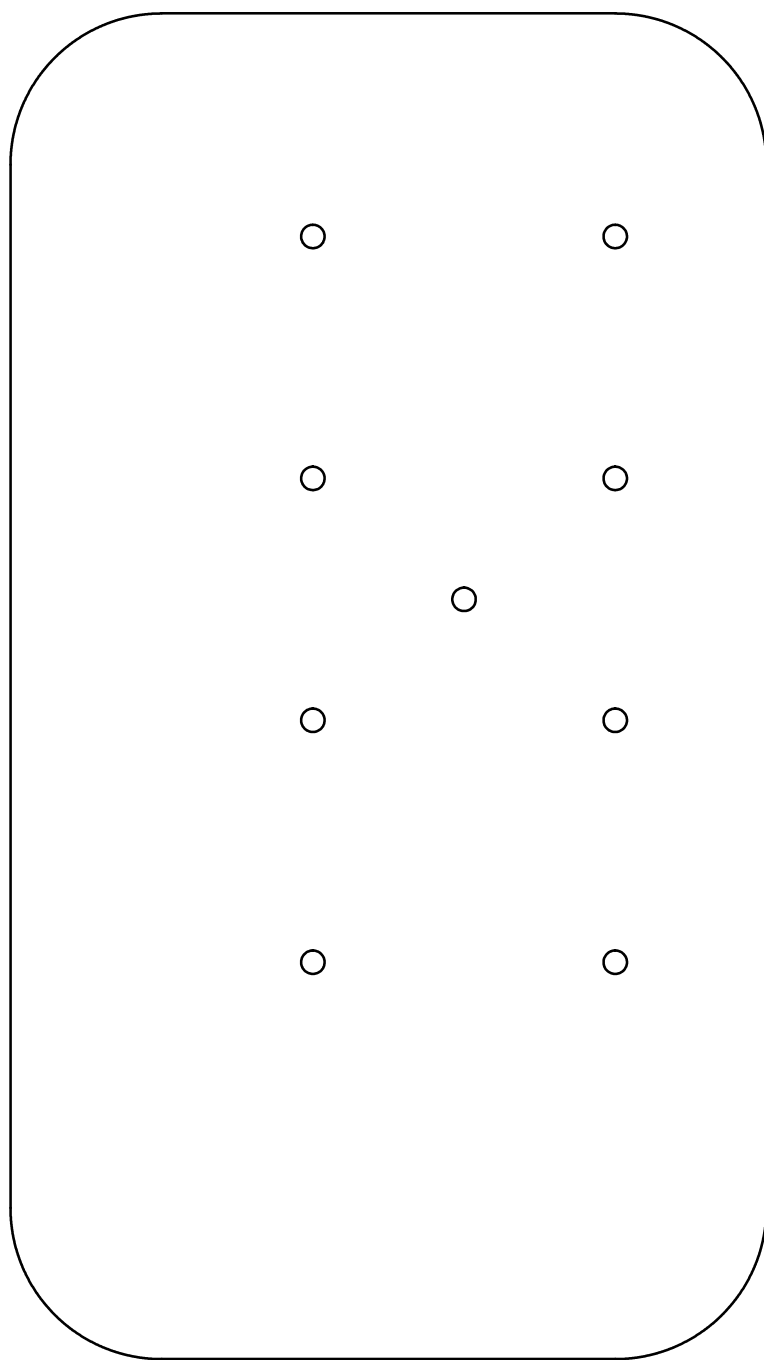


neoriginální páka ve tvaru kotoučku a její místo v kyčli

P1



pánev



sněžnice