

**ZÁPADOČESKÁ UNIVERZITA V PLZNI
FAKULTA ELEKTROTECHNICKÁ**

KATEDRA APLIKOVANÉ ELEKTRONIKY A TELEKOMUNIKACÍ

BAKALÁŘSKÁ PRÁCE

Elektromechanické doplňky kolejiště

ZÁPADOČESKÁ UNIVERZITA V PLZNI
Fakulta elektrotechnická
Akademický rok: 2012/2013

ZADÁNÍ BAKALÁŘSKÉ PRÁCE
(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Jindřich HESS**
Osobní číslo: **E09B0269P**
Studijní program: **B2612 Elektrotechnika a informatika**
Studijní obor: **Elektronika a telekomunikace**
Název tématu: **Elektromechanické doplňky kolejiště**
Zadávající katedra: **Katedra aplikované elektroniky a telekomunikací**

Z á s a d y p r o v y p r a c o v á n í :

Pro modelové kolejiště KAE navrhňte a realizujte následující doplňky:

1. Náhrada přestavníků výměn pomocí serva
2. Pohon mechanických závor
3. Programovací kolej pro DCC dekodéry

Ve všech případech uvažte stávající infrastrukturu řízení kolejiště.

Rozsah grafických prací: podle doporučení vedoucího
 Rozsah pracovní zprávy: 20 - 30 stran
 Forma zpracování bakalářské práce: tištěná/elektronická
 Seznam odborné literatury:

Student si vhodnou literaturu vyhledá v dostupných pramenech podle doporučení vedoucího práce.

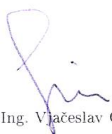
Vedoucí bakalářské práce: **Ing. Petr Weissar, Ph.D.**
 Katedra aplikované elektroniky a telekomunikací
 Konzultant bakalářské práce: **Ing. Petr Weissar, Ph.D.**
 Katedra aplikované elektroniky a telekomunikací
 Datum zadání bakalářské práce: **15. října 2012**
 Termín odevzdání bakalářské práce: **7. června 2013**



Doc. Ing. Jiří Hammerbauer, Ph.D.
 děkan



L.S.



Doc. Dr. Ing. Vjačeslav Georgiev
 vedoucí katedry

V Plzni dne 15. října 2012

Abstrakt

Předkládaná bakalářská práce popisuje realizaci přestavníků pomocí serva. V první části je shrnuta stávající konstrukce přestavníků na kolejišti KAE. V druhé části je řešeno servo jako nový typ přestavníku, je zmíněno vnitřní složení serva a ovládání pulzně šířkovou modulací (PWM). V další části jsou představeny servodekodéry, zařízení, která řídí modelářská serva. Pro vypracování této práce jsem zvolil mikrokontrolér ATmega328P, který je řešen v kapitole spolu s ostatními servodekodéry. Pomocí ATmega328P řídím přestavníky vlastní výroby, které jsou popsány v následující kapitole. V poslední části práce je teoretický popis programovací DCC koleje a zmíněné způsoby programování DCC centrálou a PC softwary. V závěru práce jsou porovnány výhody a nevýhody různých přestavníků a náročnost jejich konstrukce.

Klíčová slova

Přestavník, Memwire, servo, výhybka, závory, PWM, dekodér, Arduino, kolej, program

Abstract

This thesis describes the manipulator realization by servo. The first section summarizes the current design of manipulators in railway at KAE. The second section deals with new servo manipulator. The inner construction of servo is mentioned and the Pulse Width Modulation control is presented. The next section presents servodecoders to control the servo. I have used ATmega328P microcontroller for completing this thesis. ATmega328P is described with other servodecoders in the following chapter. The last part of this thesis describes the theory of DCC Programming track and possible programming ways by DCC central switchboard and PC softwares. At the end of this thesis there are compared manipulator advantages and disadvantages of different kinds. And difficulty of construction.

Key words

Manipulator, Memwire, servo, railroad switch, railroad barrier, PWM, decoder, Arduino, rail (track), program

Prohlášení

Předkládám tímto k posouzení a obhajobě bakalářskou práci, zpracovanou na závěr studia na Fakultě elektrotechnické Západočeské univerzity v Plzni.

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně, s použitím odborné literatury a pramenů uvedených v seznamu, který je součástí této bakalářské práce.

Dále prohlašuji, že veškerý software, použitý při řešení této bakalářské/diplomové práce, je legální.

V Plzni dne 7.6.2013

Jméno příjmení

.....

Obsah

| | |
|---|-----------|
| ÚVOD | 8 |
| SEZNAM SYMBOLŮ | 9 |
| 1 PŘESTAVNÍKY | 10 |
| 1.1 MEMORY WIRE..... | 10 |
| 1.1.1 Tvarová paměť | 10 |
| 1.1.2 Pro-Rail Wire..... | 11 |
| 1.1.3 Konstrukce | 12 |
| 1.1.4 Cena..... | 12 |
| 1.2 SERVO | 13 |
| 1.2.1 Stejnoseměrný motor | 14 |
| 1.2.2 Převodovka | 15 |
| 1.2.3 Řídící elektronika..... | 15 |
| 1.2.4 PWM – Pulzně šířková modulace | 16 |
| 2 OVLÁDÁNÍ SERVO PŘESTAVNÍKŮ | 18 |
| 2.1 TOVÁRNÍ SERVODEKODÉRY | 18 |
| 2.1.1 ESU SwitchPilot servo | 19 |
| 2.1.2 DCCkoleje dekodér pro serva..... | 19 |
| 2.2 VÝVOJOVÁ DESKA ARDUINO UNO..... | 19 |
| 2.2.1 ATmega328P..... | 20 |
| 2.2.2 Vývojové prostředí | 20 |
| 2.2.3 Knihovna Servo..... | 21 |
| 3 VLASTNÍ NÁVRH A REALIZACE PŘESTAVNÍKŮ..... | 22 |
| 3.1 MECHANICKÁ ČÁST | 22 |
| ELEKTRICKÁ ŘÍDÍCÍ ČÁST | 25 |
| 4 PROGRAMOVACÍ KOLEJ PRO DCC DEKODÉRY..... | 26 |
| 4.1 LOKODEKODÉRY..... | 26 |
| 4.1.1 Nastavení, programování..... | 26 |
| ZÁVĚR | 28 |
| POUŽITÁ LITERATURA..... | 29 |
| PŘÍLOHY..... | 1 |

Úvod

Předkládaná práce se zabývá novou konstrukcí přestavníků.

Náhrada servy je nutná, protože přestavníky s Memory Wire zabírají poměrně hodně místa a jsou pomalejší. Serva jsou oproti Memory Wire menší, silnější a rychlejší.

Pro návrh ovládání serva jsem našel na webových stránkách ostatních modelářů jeden způsob, a to pomocí servodekodérů. Jedná se o způsob programování CV registrů komerčními servodekodéry a programy v PC.

Jako zajímavější způsob jsem zvolil programování mikrokontroleru ATmega328P ve vývojové desce Arduino.

Mimo náhrady přestavníků servy, jsou v práci navrženy a realizovány mechanické závory, které jsou stahovány také servem.

Seznam symbolů

| | |
|------------|---|
| PWM | Pulse Width Modulation |
| Ni | Nikl |
| Ti | Titan |
| mm, cm, m | jednotky délky |
| Ω/m | jednotka měrného odporu |
| mA | [miliAmper], jednotka elektrického proudu |
| V | [Volt], jednotka elektrického napětí |
| Ω | [Ohm], jednotka elektrického odporu |
| mW | [miliWatt], jednotka elektrického výkonu |
| P | elektrický výkon |
| R | rezistivita, elektrický odpor |
| I | elektrický proud |
| Kč | Koruna česká |
| MW | Memory Wire |
| d | průměr |
| Ks | kus |
| PM | permanentní magnet |
| UH, UL | vysoká a nízká logická úroveň napětí („0“ a „1“) |
| T | časová perioda |
| U | elektrické napětí |
| AC, DC | Alternating (střídavý) Current (proud) a Direct (stejnoseměrný) Current (proud) |
| J-K | sběrnice pro jízdu na kolejišti |
| PC | počítač |
| CV | registr v dekodéru |
| M-S | Master Slave |
| Rx, Tx | přijímaný a vysílaný signál sériového portu |
| KHz | [kiloHertz], jednotka frekvence |
| KB | [kilobyte], datová jednotka |

1 Přestavníky

Přestavník je zařízení, které umožňuje změnu polohy, v případě této práce změnu polohy výhybek a mechanických závor. Elektrické a mechanické části konkrétních přestavníků jsou řešeny v následujících kapitolách.

1.1 Memory Wire

Memory Wire (dále jen Memwire) je "drátek s pamětí" z materiálu s tvarovou pamětí (SMA – Shape Memory Alloy). Takový drátek se průchodem elektrického proudu zahřeje a na rozdíl od běžného drátku se vlivem tepla stáhne, a proto je možné využít této vlastnosti k přestavení polohy nějakého pohyblivého dílu překvapivě velkou silou. Po vypnutí elektrického proudu se drátek vrátí (prodlouží) do své původní polohy (délky) a s ním i přestavovaný mechanismus.¹

1.1.1 Tvarová paměť

Slitiny s tvarovou pamětí mění svoji krystalickou strukturu v závislosti na teplotě. Jev tvarové paměti je bezdifuzní fázová přeměna v pevném stavu, nazývána martenzitická transformace. Průběh transformace lze řídit změnou teploty nebo vnějšího napětí. Fáze, existující při vyšší teplotě, má kubickou krystalovou mřížku a nazývá se **austenit**. Fáze, vzniklá ochlazením nebo působením vnějších sil, se nazývá **martenzit** a má krystalovou mřížku s nižší symetrií. Pro opakovanou změnu struktury je třeba působit vnější silou, a proto taková tvarová paměť je jednosměrná.²

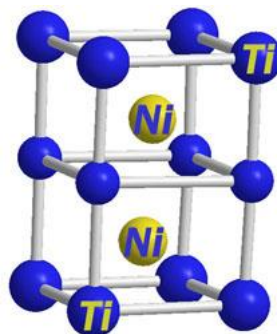
Pro dosažení dvousměrného efektu je nutné zdeformovat materiál už při výrobě za zvláštních podmínek. A kovová struktura si následně pamatuje dva tvary – původní tvar a tvar definovaným způsobem deformovaný. Předmět, vyrobený z materiálu s dvousměrným efektem, mění tvar podle teploty a přechází z tvaru výchozího do deformovaného a zpět i bez zásahu vnějších sil. Jedná se o vratný děj. Pro změnu vlastností tvarové paměti je možné vyrábět slitiny z různých kovů. Slitiny s tvarovou pamětí lze dodávat v různých podobách –

¹ LITOMYSKÝ, Petr. MEMWIRE. *Litomyský.cz* [online]. 2004 [cit. 2013-06-02]. Dostupné z: <http://www.litomysky.cz/txt/MEMWIRE.PDF>

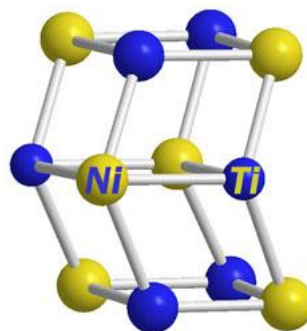
² Martenzitická transformace. *Fyzikální ústav AV ČR, v.v.i.: Oddělení funkčních materiálů* [online]. 2013 [cit. 2013-06-02]. Dostupné z: http://ofm.fzu.cz/sma/brana_cz/info/mtransformace.htm

tyče, trubky, pružiny apod. Pro modelářské účely je to drátek ze slitiny Niklu a Titanu (NiTi).³

Obrázky *Obr. 1.1* a *Obr. 1.2*, které ukazují rozdíl mezi strukturami, jsou zobrazeny níže.⁴ Obrázek, popisující jednosměrný a dvousměrný efekt, je přiložen v **příloze A** (navíc se nachází v dokumentu MEMWIRE.pdf, který je přiložen ve složce *Podklady* s elektronickou podobou této Bakalářské práce).



Obr. 1.1 Kubická krystalová struktura austenitu slitiny



Obr. 1.2 Monoklinická krystalová struktura martensitu slitiny Ni-Ti

1.1.2 Pro-Rail Wire

Modelářský Memwire má svůj název Pro-Rail Wire. Jedná se o tenký drátek s významným elektrickým odporem a pevností srovnatelnou s ocelovou strunou. Nejsnazší metoda zahřátí tohoto drátku je Jouleovým teplem, vznikajícím při průchodu elektrického proudu (jedno jestli stejnosměrného nebo střídavého). Pro účely modelářských přestavníků (opakované změny poloh) je Pro-Rail vyroben z materiálu s dvousměrnou tvarovou pamětí.

³ LITOMYSKÝ, Petr. MEMWIRE. *Litomyský.cz* [online]. 2004 [cit. 2013-06-02]. Dostupné z: <http://www.litomysky.cz/txt/MEMWIRE.PDF>

⁴ Martenzitická transformace. *Fyzikální ústav AV ČR, v.v.i.: Oddělení funkčních materiálů* [online]. 2013 [cit. 2013-06-02]. Dostupné z: http://ofm.fzu.cz/sma/brana_cz/info/mtransformace.htm

Pro-Rail o průměru 0.12 mm má měrný odpor 75 [Ω/m] (ve studeném stavu). Pro plné využití paměťového jevu jím musí procházet elektrický proud o velikosti cca 200 mA (při doporučených 3V napájecího napětí). Pro stavbu přestavníku je obvykle potřeba 50 – 150 mm drátku (délka drátku závisí na požadovaném zdvihu), odpor tedy činí 4 – 12 Ω a potřebný elektrický příkon je 160 – 480 mW.⁵ Takovéto energetické zatížení zvládne každý zdroj pro napájení železnice. Uvedené hodnoty příkonu jsou získány z rovnice:

$$P = R \cdot I^2 \quad (1.1)$$

1.1.3 Konstrukce

Pro-Rail jedním koncem připevníme na rám kolejiště a druhým k pohyblivé části přestavovaného mechanismu (v případě stávajícího kolejiště na KAE k táhlu výhybky). Pro-Rail umožňuje pozvolný tichý lineární pohyb (bez převodů) a v přestavené poloze se sám zastaví na rozdíl od elektromagnetických přestavníků. Po připojení napájení se drátek zkrátí a změni polohu přestavníku jedním směrem. Po vypnutí elektrického proudu drátek vychladne vlivem okolního prostředí a vrátí se do původní polohy. Nevýhoda opětovného prodloužení drátku je ta, že přestavník nemusí dorazit do krajní polohy kvůli nedostatku tlakové síly, tuto sílu můžeme vyvinout pružinou nebo pomocí závaží. Je vhodné chránit přestavník před proudícím vzduchem, aby nedošlo k ochlazení drátku. To by vedlo k nedostatečné změně jeho délky, vedoucí například k vykolejení mašiny (v případě výhybky). Přestavná doba změny polohy přestavníku při pokojové teplotě a bez proudění vzduchu při skokové změně napájecího napětí je cca 2-3 vteřiny.⁶

1.1.4 Cena

Samotný drátek Memwire vyjde na cenu 375 Kč/m při zvýhodněné ceně za základní délku drátu 1.2 m (450 Kč). Následující tabulka ukazuje ceny kompletního příslušenství Memwire včetně drátu samotného:

⁵ LITOMYSKÝ, Petr. MEMWIRE. *Litomyský.cz* [online]. 2004 [cit. 2013-06-02]. Dostupné z: <http://www.litomysky.cz/txt/MEMWIRE.PDF>

⁶ LITOMYSKÝ, Petr. Přestavník. *Litomyský.cz* [online]. 2001 [cit. 2013-06-02]. Dostupné z: <http://www.litomysky.cz/mat/mwprestavnik.htm>

Tab. 1.1 Ceny Memwire a jeho příslušenství

| č. kategorie | výrobek | cena [Kč] |
|-----------------|--|--------------|
| MW 51m | teflonová bužírka pro memory wire - metráž (cena za metr platná pro všechny délky kromě 0.6m, nejmenší odběr 0.25 m) | 90 |
| MW 52m | teflonová bužírka pro memory wire - 0.6 metru (základní délka za zvýhodněnou cenu) | 40 |
| MW 60 | svorky pro memory wire, 20 ks | 30 |
| MW 70 | pružinky pro memory wire, 10 ks | 60 |
| MW 80 | očka (pro šroub M2) pro memory wire, 20 ks | 30 |
| MW 11m | drát s tvarovou pamětí $d = 0.125\text{mm}$ (memory wire) - metráž (cena za metr pro všechny délky kromě 1.2m a 25m, nejmenší odběr 0.6 m) | 410 |
| MW 12m | drát s tvarovou pamětí $d = 0.125\text{mm}$ (memory wire) - 1.2 metru (základní délka za zvýhodněnou cenu) | 450 |
| MW TK | memory wire - základní sada | 570 |
| MW 90 | přestavník s pamětovým drátkem, napájecí napětí 5V, hotový | 290 |
| MW 90s | přestavník s pamětovým drátkem, napájecí napětí 5V, stavebnice | 240 |

Z tabulky tedy vyčísím cenu jednoho přestavníku pro jednoduchou výhybku (za zvýhodněné ceny), při výrobě použiji odhadem: 10 cm Memwire (MW), 10 cm teflonové bužírky (TB), 1 svorku (S), 1 pružinku (P) a 2 očka (O). Vše sečtu v následující rovnici:

$$\begin{aligned} \text{Cena} &= MW + TB + S + P + 2 \cdot O = & (1.2) \\ &= 450/12 + 40/6 + 30/20 + 60/10 + 2 \cdot (30/20) = 55\text{Kč} \end{aligned}$$

Základní cena bez táhla (trnu) výhybky vyčíslená pouze z výše uvedené tabulky odpovídá přibližně 55 Kč na jeden přestavník.⁷

1.2 Servo

Servo (z latiny Servus – otrok) je zkrácený název pro servomotor, pro modelářské účely pak modelářské servo. Na rozdíl od ostatních motorů lze u serva nastavit přesnou polohu natočení osy a to je pro modelářské účely nezbytné.

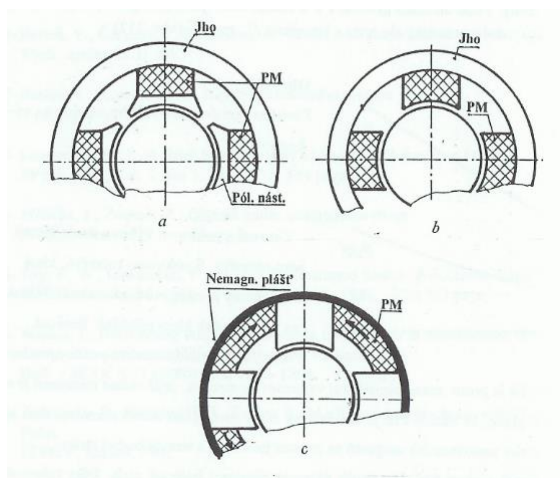
Serva jsou obecně méně hlučná než například elektromagnetické a ostatní elektromotorické přestavníky, ale i mezi samotnými servy jsou znatelné rozdíly v hlučnosti pracovního chodu. To pochopitelně závisí na kvalitě a ceně samotných serv.

⁷ LITOMYSKÝ, Petr. Ceník Memwire. *Litomyský.cz* [online]. 2001 [cit. 2013-06-02]. Dostupné z: <http://www.litomysky.cz/mat/mwcceny.htm>

Modelářské servo je složeno ze tří základních částí: stejnosměrného motoru, převodovky a řídicí elektroniky.

1.2.1 Stejnosměrný motor

Pro servopohony se používá stejnosměrný motor s permanentními magnety (dále jen PM). Čím menší rozměry motoru, tím je výhodnější použít buzení s PM. PM zabírají podstatně méně místa než budicí vinutí a tímto odpadají ztráty buzením. „Kotva (rotor) může mít klasické provedení s komutátorem, pak budicí permanentní magnety jsou umístěny na statoru.⁸“ Na Obr. 1.3 jsou zobrazena různá uspořádání PM:



Obr. 1.3 Budicí PM u stejnosměrného stroje⁹

- a – radiálně uspořádané magnety
- b – totéž bez pólových nástavců
- c – segmentové magnety s naznačenou magnetizací

„Uspořádání c vzhledem k větší délce PM dovolují jejich menší průřez.¹⁰“

⁸ BARTOŠ, Václav. *Elektrické stroje*. Vyd. 1. Plzeň: Západočeská univerzita, 1995, s. 199-200. ISBN 80-7082-221-x.

⁹ BARTOŠ, Václav. *Elektrické stroje*. Vyd. 1. Plzeň: Západočeská univerzita, 1995, s. 199-200. ISBN 80-7082-221-x.

¹⁰ BARTOŠ, Václav. *Elektrické stroje*. Vyd. 1. Plzeň: Západočeská univerzita, 1995, s. 199-200. ISBN 80-7082-221-x.

1.2.2 Převodovka

V převodovce jsou uloženy převody – ozubená kola, která slouží k nastavení požadovaného poměru rychlosti a síly. V převodovce je také k hřídeli posledního ozubeného kola připevněn zpětnovazební potenciometr.

Kvalita převodovky je dána materiálem použitým pro výrobu převodů. Mohou být plastové nebo kovové. To se následně odráží na váze a ceně samotného serva.

1.2.3 Řídící elektronika

Obecně jsou v servo systému tři základní obvody (smyčky):

1. Proudová smyčka
2. Rychlostní smyčka
3. Polohovací smyčka

Proudová smyčka je velmi rychlý vnitřní obvod, který řídí proud do vinutí motoru, a tím přímo ovlivňuje točivý moment motoru.

Rychlostní smyčka je nadřazená proudové smyčce a je pomalejší. Snímá údaj z externího senzoru, např. stejnosměrného tachogenerátoru (údaj o skutečné rychlosti) a porovnává ho s požadovanou hodnotou, která může pocházet z potenciometru, řídicího nebo polohovacího systému. „*Touto kombinací se získá rychlostní servo pro přesnou regulaci ve velkém regulačním rozsahu.*“¹¹

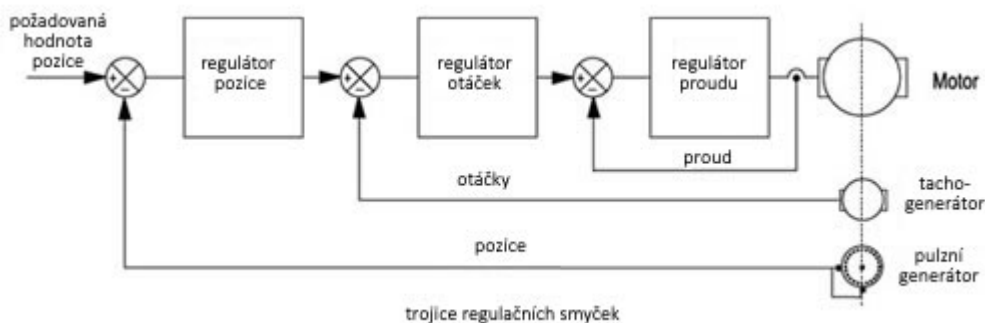
Polohovací smyčka porovnává skutečnou hodnotu z pozičního senzoru (např. polohu zpětnovazebního potenciometru) s požadovanou hodnotou (z řídicího signálu PWM) a řídí proudovou a rychlostní smyčku aby bylo dosaženo minimální odchylky.¹²

„*Obrázek níže ukazuje princip kompletního pozičního systému se servomotorem. Některé systémy minou rychlostní smyčku, řídí moment motoru a ne otáčky. Toho se dosáhne řízením proudu místo řízení otáček (točivý moment je úměrný proudu).*“¹³

¹¹ Škola - Všeobecně o Servo Technice. *Pohonnatechnika.cz* [online]. 2007, 2013 [cit. 2013-06-03]. Dostupné z: <http://www.pohonnatechnika.cz/skola/servo/skola-vseobecne-o-servo-technice>

¹² Škola - Všeobecně o Servo Technice. *Pohonnatechnika.cz* [online]. 2007, 2013 [cit. 2013-06-03]. Dostupné z: <http://www.pohonnatechnika.cz/skola/servo/skola-vseobecne-o-servo-technice>

¹³ Škola - Všeobecně o Servo Technice. *Pohonnatechnika.cz* [online]. 2007, 2013 [cit. 2013-06-03]. Dostupné z: <http://www.pohonnatechnika.cz/skola/servo/skola-vseobecne-o-servo-technice>



Obr. 1.4 Kompletní poziční systém se servomotorem¹⁴

Serva existují analogová a digitální, oba typy využívají vstupní řídicí PWM signál (PWM řízení je řešeno v kapitole *PWM – Pulzně šířková modulace*). Rozdíl mezi servy je v analogové (speciální obvod) a digitální (mikroprocesor) řídicí elektronice. Digitální serva dokáží pracovat s vyššími frekvencemi impulsů. Jsou silnější a přesnější, ale mají vyšší spotřebu než analogová serva stejné kategorie (velikost, použití, atd.). U některých digitálních serv lze naprogramovat například rychlost pohybu, smysl otáčení, neutrální polohu a koncové body.

1.2.4 PWM – Pulzně šířková modulace

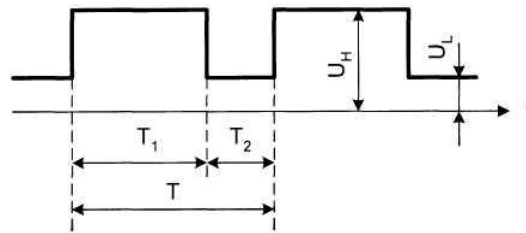
Modulátorem PWM je vybaven téměř každý mikrokontrolér, a proto pro ovládání přestavníků v rozsahu této práce využívám ATmega328P, jak je popsáno v kapitole *Vývojová deska Arduino UNO*.

Pulzně šířková modulace (PWM – Pulse-Width Modulation) je vhodná pro přenos signálů na větší vzdálenosti v přítomnosti rušení. Využívá se právě pro ovládání elektromotorů, elektromagnetů, apod. Řízení PWM vychází z řízení střední hodnotou napětí výstupního signálu, ten je tvořen hodnotami U_H (log. „1“) a U_L (log. „0“). Střední hodnota napětí je dána vztahem:

$$U_{stř} = \frac{T_1}{T} (U_H - U_L) + U_L \tag{1.3}$$

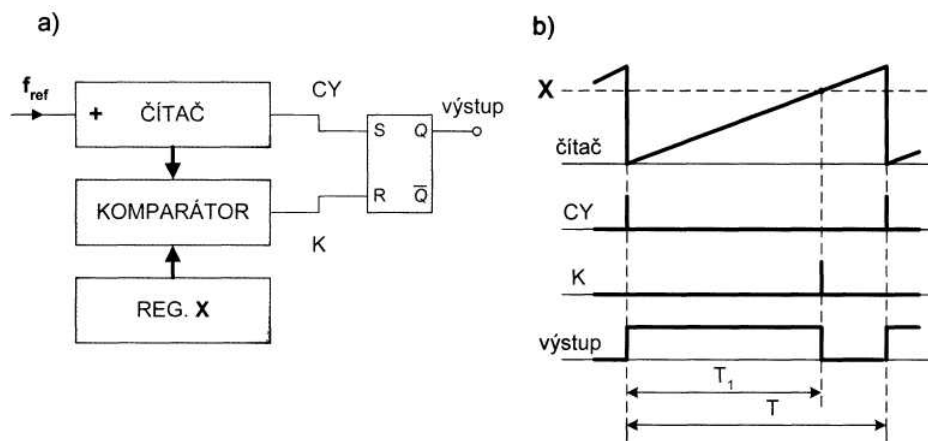
podle Obr. 1.5.

¹⁴ Škola - Všeobecně o Servo Technice. *Pohonnatechnika.cz* [online]. 2007, 2013 [cit. 2013-06-03]. Dostupné z: <http://www.pohonnatechnika.cz/skola/servo/skola-vseobecne-o-servo-technice>



Obr. 1.5 Signál s PWM¹⁵

PWM modulátor je založen na čítači, ten počítá nahoru a při přechodu ze stavu 11...1 do 00...0 překlopí klopný obvod do stavu 1. Součástí modulátoru je dále registr a komparátor. V registru je nastavena hodnota a ta je porovnávána komparátorem s hodnotou v čítači. Při shodě hodnot je klopný obvod vynulován. Nejjednodušší zapojení PWM modulátoru a odpovídající průběh signálů ukazují Obr. 1.6a a Obr. 1.6b.¹⁶



Obr. 1.6 Modulátor PWM¹⁷

¹⁵ PINKER, Jiří. *Mikroprocesory a mikropočítače*. 1. vyd. Praha: BEN - technická literatura, 2004, s. 111-114. ISBN 80-730-0110-1.

¹⁶ PINKER, Jiří. *Mikroprocesory a mikropočítače*. 1. vyd. Praha: BEN - technická literatura, 2004, s. 111-114. ISBN 80-730-0110-1.

¹⁷ PINKER, Jiří. *Mikroprocesory a mikropočítače*. 1. vyd. Praha: BEN - technická literatura, 2004, s. 111-114. ISBN 80-730-0110-1.

2 Ovládání servo přestavníků

Ovládat serva lze různými způsoby. Slouží k tomu elektronická zařízení, která se nazývají obecně servodekodéry.

Servodekodéry se od sebe liší: **způsobem ovládání** (analogovým a digitálním), **výstupními funkcemi** (serva) a **způsobem programování**.

Způsob ovládání

1. Digitální sběrnice J-K – Výstupům lze přidělit určité adresy a následně je ovládat z ovladače nebo pomocí softwaru v PC.
2. Kontakty pro přímé ovládání (například tlačítkovými spínači, relé, apod.)

Výstupní funkce

1. Rychlost přeběhu – všechny výstupy najednou, jednotlivé výstupy, různá rychlost pro každý směr
2. Krajiní polohy
3. Reverzovní chod – jednoduše i prohozením krajiní poloh
4. Speciální funkce – vazba na jiný výstup nebo zákmity v krajiní poloze

Programování

1. Programovacím tlačítkem – adresa, další parametry
2. Programováním CV registrů (tyto registry jsou podrobněji řešeny v kapitole)
3. Oba způsoby¹⁸

Většinou bývá u servodekodérů více než jeden výstup na serva (4, 8 výstupů). Od tohoto se obvykle odvíjí cena na jeden výstup (více výstupů vychází cenově lépe).¹⁹

V předchozí kapitole *Servo* jsem nastínil vnitřní složení serva a obecnou strukturu řídicích obvodů. V této kapitole uvádím některé „tovární“ servodekodéry a vlastní ovládání pomocí mikroprocesoru ATmega328P na vývojové desce Arduino UNO. A tímto bych uvedl čtvrtý způsob **programování** – vlastní.

2.1 Tovární Servodekodéry

¹⁸ PINTA, Martin. Servodekodéry. *Lokopin* [online]. 2006, 2013 [cit. 2013-06-04]. Dostupné z: <http://lokopin.wz.cz/digital/servdec.htm>

¹⁹ PINTA, Martin. Servodekodéry. *Lokopin* [online]. 2006, 2013 [cit. 2013-06-04]. Dostupné z: <http://lokopin.wz.cz/digital/servdec.htm>

2.1.1 ESU SwitchPilot servo

ESU SwitchPilot servo je dekodér od firmy ESU (Electronic Solutions Ulm). Lze ho připojit na napájení 16 V (zvádné až 18 V AC) a na sběrnici J-K (DC), po které jdou ovládací signály. Má 4 výstupy na serva, je programovatelný přes tlačítka i přes CV registry. Serva lze spouštět i bez použití digitálu pouze pomocí spínacích tlačítek. Tento dekodér umožňuje nastavení rychlosti i každé krajní polohy zvlášť. Proto je možné, prohozením krajních poloh, nastavení i reverzního pohybu.

Cena dekodéru se pohybuje od 600 Kč do 750 Kč.

2.1.2 DCCkoleje dekodér pro serva

Tento dekodér pochází ze stránek [HTTP://WWW.DCCKOLEJE.KX.CZ/SERVO.HTML](http://www.dcckoleje.kx.cz/servo.html). Vyrábí se a dodává ve variantě Master-Slave (Master má usměrňovač). Každý modul (M i S) může ovládat 8 serv, sestava Master-Slave tedy umožní ovládat až 16 serv najednou.

Programování adresy přes tlačítka je skoro stejné jako u ESU, jen je nutné tlačítka držet i při zadání příkazu pro přestavení.²⁰

2.2 Vývojová deska Arduino UNO

Arduino je otevřená (volně použitelná) vývojová platforma založená na všestranném nenáročném hardwaru a softwaru.

K vlastnímu ovládání přestavníků jsem zvolil, pro svoji dostupnost, vývojovou desku Arduino UNO R3. Je to deska s mikroprocesorem ATmega328P.

Deska obsahuje (s ohledem na parametry ATmega328P):

- 14 digitálních vstupů/výstupů (6 z nich může být použito jako PWM výstupy), piny 0 a 1 slouží pro sériovou komunikaci (označeny Rx a Tx)
- Dvě LED indikující sériový přenos (označeny RX a TX)
- 6 analogových vstupů
- 16 MHz keramický oscilátor (tvořen krystalem)
- USB konektor (standardní napájení 5V z PC)

²⁰ PINTA, Martin. Servo a výhybky. *Lokopin* [online]. 2006, 2012 [cit. 2013-06-04]. Dostupné z: <http://lokopin.wz.cz/kolejivo/servo.htm>

- Napájecí jack konektor (doporučené napájení 7 až 12 V, zvládne až 20 V → může se přehřívat a hrozí poškození desky)
- 5V pin, 3.3V pin (max. proud 50 mA), Vin (vstupní napájení) pin, IOREF pin (umožňuje ochranám přizpůsobit se napětí poskytovaného deskou) a 3x GND (zemnicí) piny
- ICSP konektor
- Resetovací tlačítko
- Obvody potřebné pro chod mikroprocesoru
- Mikroprocesor Atmega16U2 naprogramovaný jako USB ↔ USART konvertor

Fotografie vývojové desky Arduino UNO přiložena v **příloze B**.

2.2.1 ATmega328P

Srdcem vývojové desky Arduino UNO je výše zmiňovaný mikroprocesor ATmega328P. Parametry (ve stručnosti) samotného mikroprocesoru jsou:

- Jedná se o 8 bitový mikroprocesor s RISC architekturou
- Až 20 KHz pracovní frekvence (použito 16 KHz)
- 3 2KB flash paměť
- 1 KB EEPROM paměť
- 2 KB vnitřní SRAM paměť
- Dva 8 bitové Čítače/Časovače a jeden 16 bitový Čítač/Časovač
- 6 PWM kanálů
- Programovatelný sériový USART

2.2.2 Vývojové prostředí

Desku Arduino lze programovat ve vývojovém prostředí založeném na processingu²¹ (uživatel vytváří tzv. sketch) v jazyce wiring²². Vývojové prostředí je zdarma ke stažení stránkách Arduina²³, snadno se ovládá a po správném nainstalování ovladačů Arduina ani

²¹ <http://www.processing.org/>

²² <http://wiring.org.co/>

²³ <http://www.arduino.cc/>

není nutné prakticky cokoliv nastavovat. Jak na stránkách arduina, tak ve vývojovém prostředí, jsou příklady (ukázky) programů k rozmanitým funkcím a knihovnám arduina. Vytvořený sketch lze jedním kliknutím zkompileovat a odeslat do vývojové desky.

Pro jednoduchost jsem zvolil toto vývojové prostředí. Musím ovšem dodat, že ovladače Arduina nelze tak snadno nainstalovat na Windows 8 jako například na Windows 7 nebo Windows Vista (z vlastní zkušenosti jsem zkoušel instalaci na Win Vista a následně na Win 8). Na internetu jsem našel návod, jak instalaci úspěšně provést a prý je to možné dokonce i více způsoby. Podle návodu byl problém úspěšně vyřešen.

2.2.3 Knihovna Servo

Pro ovládání servo přestavníku stačí jediná knihovna funkcí – **Servo**. Tato knihovna umožňuje na většinu desek Arduino připojit až 12 modelářských serv najednou, ale zablokuje přitom funkci `analogWrite()` (PWM) na piny 9 a 10, ať už je na nich servo připojeno či nikoli.

Při napájení z USB přes PC hrozí při připojení více než dvou serv přetížení USB portu v PC a následné zničení portu.

Knihovna umožňuje použít tyto funkce:

attach() – připojí proměnnou *servo* k pinu (číslo pinu). Syntaxe: `servo.attach(pin)` nebo `servo.attach(pin, min, max)` → `min` nastavuje šířku pulzu v μs odpovídající minimálnímu úhlu (0°) natočení serva (výchozí hodnota je 544) a `max` nastavuje analogicky opak (180° , výchozí hodnota je 2400)

attached() – kontroluje, zda je proměnná *servo* připojena k pinu. Syntaxe: `servo.attached()` → vrací hodnotu `TRUE`, když je servo připojeno a `FALSE`, když není

detach() – odpojí proměnnou *servo* od pinu. Pokud jsou všechny proměnné typu *servo* odpojeny, je možné použít piny 9 a 10 pro výstup PWM pomocí `analogWrite()`.

Syntaxe: `servo.detach()`

write() – zapíše hodnotu do serva, přímo ovládá hřídel. U standardních serv, tato funkce nastaví úhel hřídele ve stupních ($0^\circ - 180^\circ$) → servo se rozjede, aby dosáhlo daného úhlu. Pro serva s kontinuální rotací, nastaví rychlost (hodnota 0 znamená plnou rychlost v jednom směru, hodnota 180 znamená rychlost ve druhém směru, a hodnota blízká 90 znamená žádný pohyb). Syntaxe: `servo.write(angle)`

writeMicroseconds() – zapíše hodnotu do serva v μs , opět přímo ovládá hřídel. U standardních serv, tato funkce nastaví úhel hřídele způsobem: hodnota 1000 znamená

plnou půlotáčku proti směru hodinových ručiček, 2000 znamená plnou půlotáčku po směru hodinových ručiček, a 1500 znamená střed (někteří výrobci nedodržují tento standard, a tak serva častěji odpovídají hodnotám mezi 700 a 2300). Syntaxe:

`servo.writeMicroseconds(μ s)`

`read()` – přečte momentální úhel natočení ($0^\circ - 180^\circ$) serva (hodnotu předanou poslední výzvě k zápisu `write()`). Syntaxe: `servo.read()`

3 Vlastní návrh a realizace přestavníků

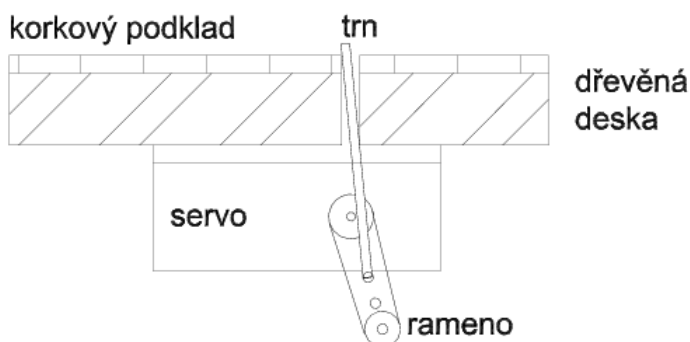
3.1 Mechanická část

K návrhu a vlastní konstrukci přestavníků jsem si vypůjčil na KAE jeden díl kolejí a jednu výhybku. Výhybka a závory jsou navrženy na dřevěné desky (tloušťky 8 mm) a korkové podklady pod koleje (tloušťky 2 mm), které pochází také přímo z kolejiště na KAE.

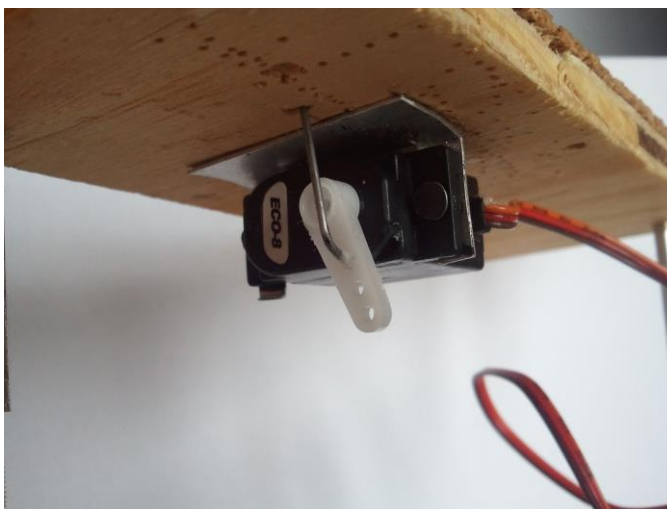
Výhybka

Posuvník výhybky je přestavován trnem (ocelový drát), který funguje jako páka. Druhý konec trnu je připevněn k pohyblivému ramenu, které je nasazeno na hřídeli modelářského mikro serva ECO 8. Trn prochází vyvrtaným otvorem v dřevěné desce a korkovém podkladu. Servo je uchyceno pomocí plechové konstrukce (tloušťky plechu necelý 1 mm), která je přilepena vteřinovým lepidlem na spodní stranu dřevěné desky. Celá konstrukce výhybky s deskou, korkovým podkladem a servem, stojí na čtyřech distančních sloupcích.

Na *Obr. 3.1* je schéma mechanické konstrukce přestavníku výhybky (výhybka není na desce zobrazena, je lépe vidět trn). Pod schématem na *Obr. 3.2* je fotografie přestavníku.



Obr. 3.1 Mechanická konstrukce výhybky



Obr. 3.2 Fotografie přestavníku výhybky

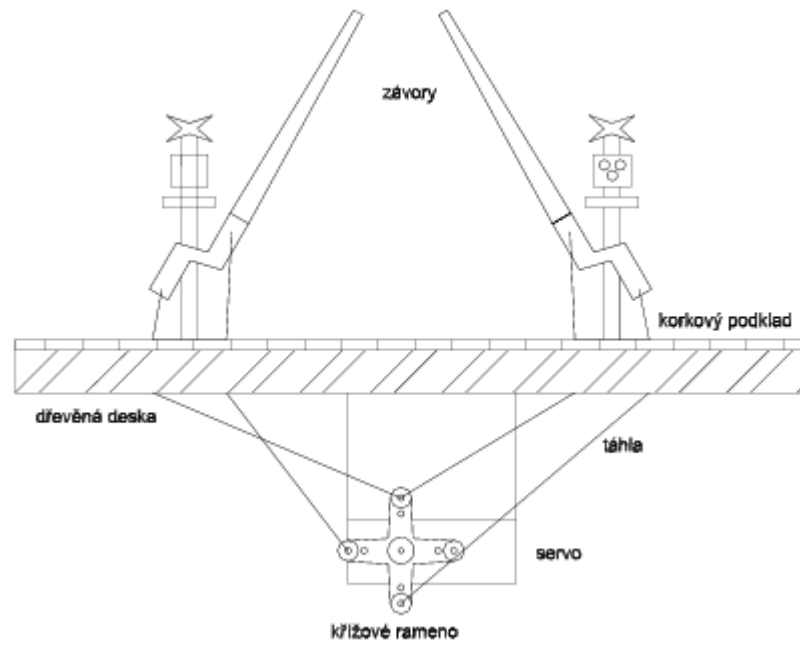
Závory

Závory jsou vyrobeny z papírové čtvrtky. Sloupek, na kterém se závory pohybují, je vyroben z dřevěné špejle (o průměru 3 mm). Velikost závor je v modelářském měřítku TT. Konstrukce závor je navržena a sestavena podle návodu na webové stránce:

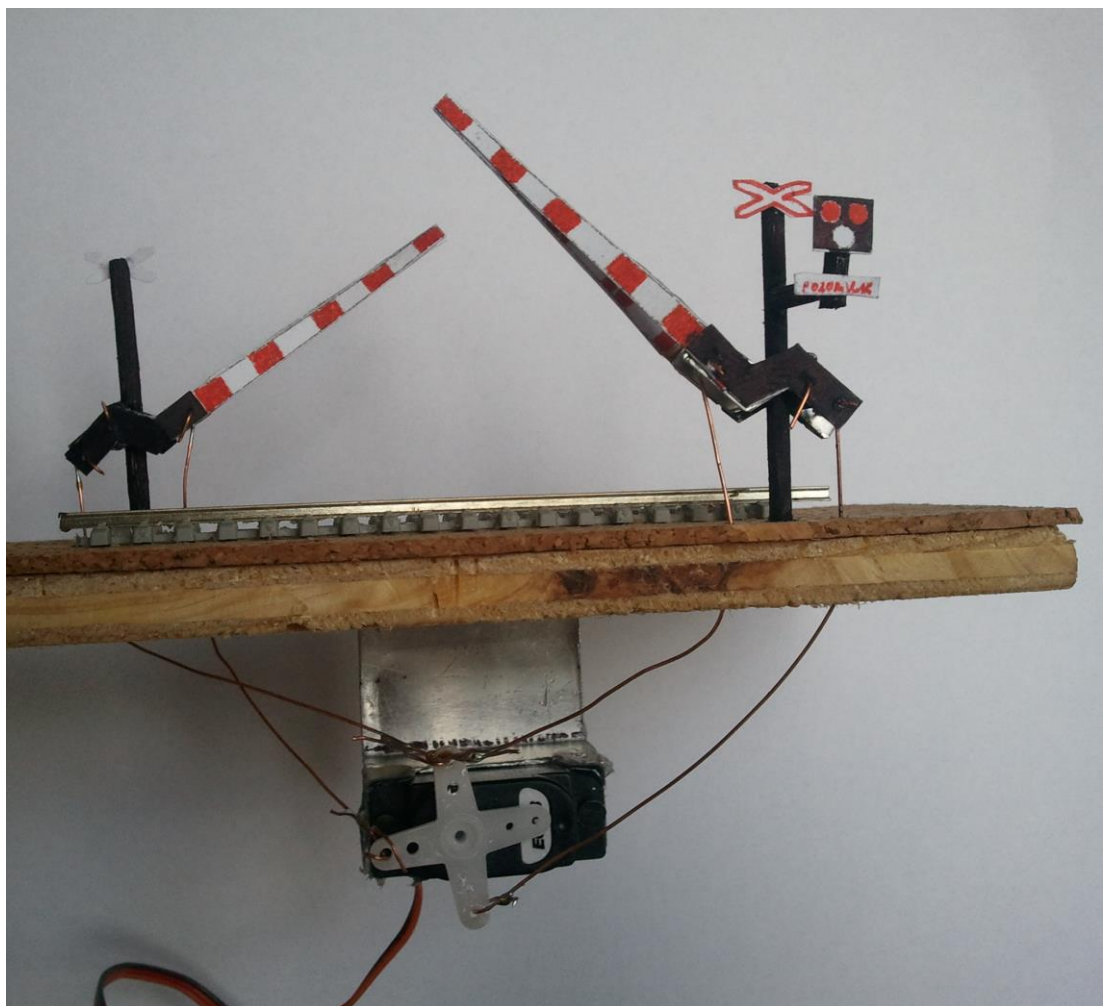
[HTTP://WWW.TTMODEL.YC.CZ/INDEX.PHP?TEMA=NAVESTIDLA](http://www.ttmodely.yc.cz/index.php?TEMA=NAVESTIDLA)

Mechanismus závor je ovládán jedním servem (také ECO 8). Servo je uchyceno podobnou plechovou konstrukcí jako u **výhybky**, navíc je přilepeno tavnou pistolí. V dřevěné desce a korkovém podkladu jsou vyvrtány otvory, kterými prochází táhla ze zvonkového drátu. Táhla jsou jedním koncem připevněna k závorám (přiletována k drátové spojce závor) a druhým koncem jsou uchycena ke křížovému ramenu, které je nasazeno na hřídeli serva. Celá konstrukce opět stojí na čtyřech distančních sloupcích.

Na *Obr. 3.3* je schéma mechanické konstrukce závor a přestavníku (koleje a distanční sloupky nejsou zobrazeny). Na *Obr. 3.4* je fotografie přestavníku a závor.



Obr. 3.3 Mechanická konstrukce závor

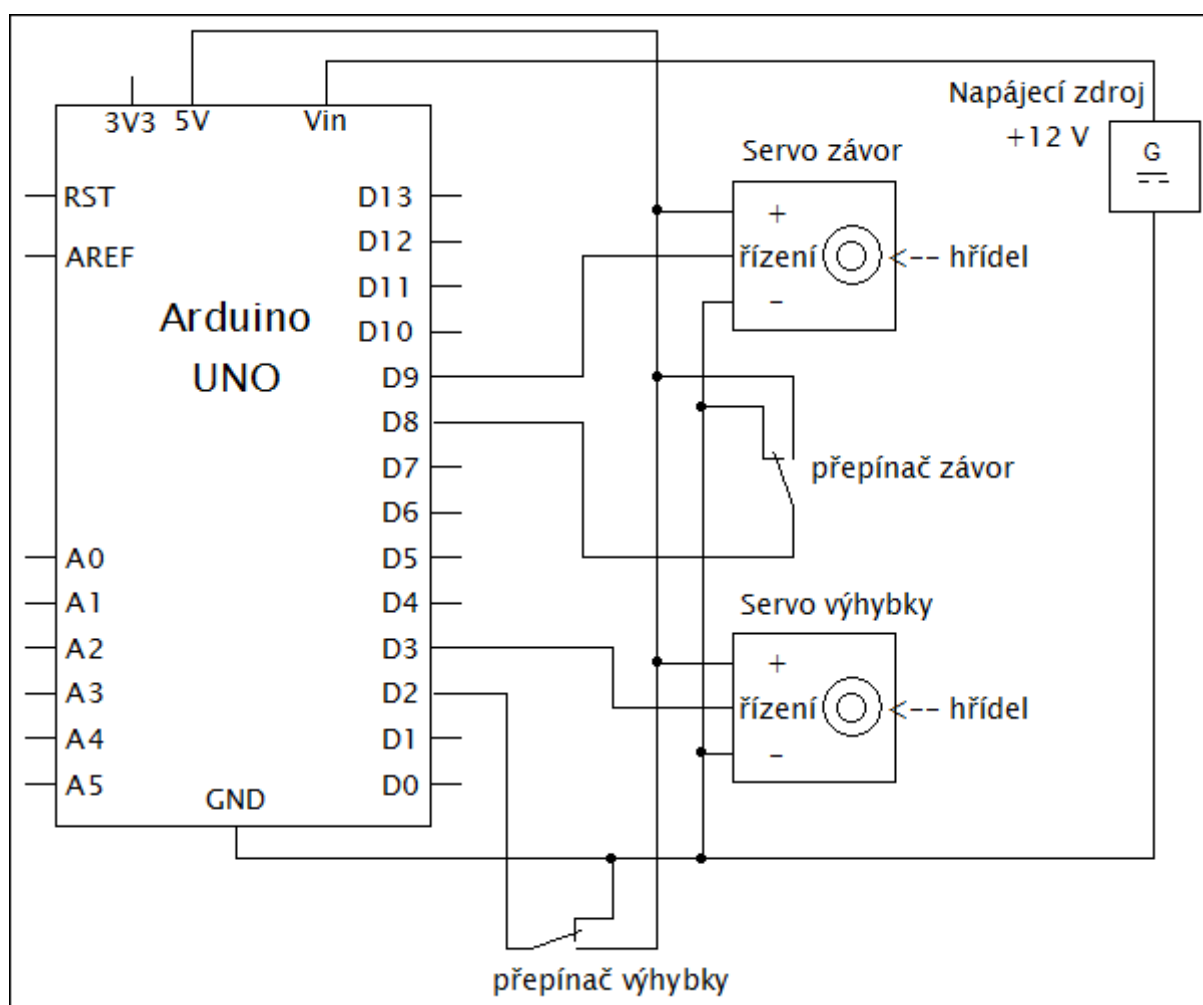


Obr. 3.4 Fotografie přestavníku závor

Elektrická řídicí část

Elektrická řídicí část je tvořena vývojovou deskou Arduino UNO. Ta může být napájena stejnosměrným zdrojem napětí (například 12 V). Na piny digitálních vstupů/výstupů 3 a 9 jsou připojena serva pro výhybku a závory. Na piny 2 a 8 jsou připojeny přepínače, které nahrazují řízení DCC centrálou. Závory zatím nejsou centrálou řízeny. Přestavníkům výhybek centrála dodává proud 200 mA jako řídicí signál (z důvodu původních Memwire přestavníků). Pro zabudování servo přestavníků výhybek do kolejiště se musí výstupní řídicí signál 200 mA převést na napěťový signál 5 V (log. „1“). To lze vyřešit tranzistorovým spínačem, který se k desce centrály připojí jako malá DPS nebo se navrhne nová část desky centrály.

Na Obr. 3.5 je schéma zapojení, vytvořené pro prezentaci.



Obr. 3.5 Schéma řízení výhybky a závora pomocí Arduino UNO

Řídicí program k uvedenému schématu je přiložen v příloze C.

4 Programovací kolej pro DCC dekodéry

Programovací kolej je izolovaná část kolejiště, na které se v jednu chvíli může nacházet pouze jedna lokomotiva (jeden lokodekodér). U továrních DCC centrál se programovací výstup (dva vodiče) na izolovanou kolej značí P a Q.

4.1 Lokodekodéry

Lokodekodér je malé elektronické zařízení, které je vloženo do modelu lokomotivy. Je napájen ze signálu z kolejí, z kterého je rovněž řízen a vytváří z něj napájení a řízení motoru, světel a dalších zařízení. Pomocí PWM nastavuje otáčky motoru. Dále plynulý rozjezd a dojezd.²⁴

4.1.1 Nastavení, programování

Lokodekodér je tvořen mikroprocesorem, a proto obsahuje velmi mnoho parametrů.

Programovatelné registry je označují CV a jsou číslovány od 1. Registry jsou osmibitové a každý obsahuje hodnotu, kterou lze, ve většině případů, měnit.

Programování registrů na programovací koleji umí DCC centrála nebo je možné CV registry programovat pomocí softwaru v PC (RailCo TrainProgramme, JMRI Decoder Pro).

Registry jsou dány normou NMRA²⁵. Část registrů je povinná, část doporučená a zbytek je volitelný (dle výrobce).

V následující tabulce jsou uvedeny důležité CV lokodekodéru Zimo MX621:

Tab. 4.1 CV pro lokodekodér Zimo MX621²⁶

| CV | Hodnota | Default | Význam | Poznámka |
|----|---|---------|-------------------|----------|
| 1 | 1 – 127 | 3 | Adresa loko | |
| 2 | 1 – 252 | 1 | Rozjezdové napětí | |
| 3 | 0 – 255 | 2 | Rozjezd | |
| 4 | 0 – 255 | 1 | Dojezd | |
| 5 | 1 – 252 | 1 | Max. rychlost | |
| 6 | $\frac{1}{4}$ až $\frac{1}{2}$ hodnoty CV5 | 1 | Střední rychlost | |
| 7 | | | Číslo verze | |

²⁴ PINTA, Martin. Lokodekodéry. *Lokopin* [online]. 2006, 2013 [cit. 2013-06-07]. Dostupné z: <http://lokopin.wz.cz/digital/lokodekodery.htm>

²⁵ http://www.nmra.org/standards/DCC/standards_rps/rp922.html

²⁶ PINTA, Martin. Lokodekodéry. *Lokopin* [online]. 2006, 2013 [cit. 2013-06-07]. Dostupné z: <http://lokopin.wz.cz/digital/lokodekodery.htm>

| | | | | |
|----|--|----------------|-------------------------|---|
| 8 | | 145 (=Zimo) | Identifikace výrobce | CV8 = „8“ → HARD RESET (všechna CV se nastaví na defaultní hodnoty) |
| 29 | | 14 | Konfigurace | |

Popis jednotlivých registrů:

CV1 – Adresa lokomotivy. Slouží k identifikaci, každá lokomotiva má svoji adresu

CV2 – Rozjezdové (startovací napětí). První bod (počátek) třibodové křivky nastavení rychlosti.

CV3 – rozjezd, akcelerace. Při zvýšení rychlosti jde o dobu, za jakou lokomotiva zareaguje. Hodnota „0“ znamená okamžitou změnu rychlosti a vyšší hodnoty vedou k plynulejší změně rychlosti.

CV4 – dojezd, decelerace. Analogicky k akceleraci.

CV5 – maximální rychlost. Tato rychlost by měla odpovídat skutečné rychlosti pravé lokomotivy, je to třetí bod křivky.

CV6 – střední rychlost. Druhý bod křivky, ovladač přesně na polovině. Je to rychlost lokomotivy při zapnuté funkci Posun.

CV7 – číslo verze firmwaru. Jedná se o program v lokodekodéru. Tento CV lze pouze číst.

CV8 – identifikační číslo výrobce. Opět je možné pouze číst.

CV29 – konfigurace. Převážně jízdních vlastností.²⁷

²⁷ PINTA, Martin. Lokodekodéry. *Lokopin* [online]. 2006, 2013 [cit. 2013-06-07]. Dostupné z: <http://lokopin.wz.cz/digital/lokodekodery.htm>

Závěr

V práci jsou shrnuty různé způsoby realizace přestavníků ostatních modelářů. Ale většina využívá k ovládní serv tovární servodekodéry.

Realizace přestavníku výhybky pomocí serva, ovládaného vývojovou deskou Arduino, byla, podle návrhu, zdařile provedena.

U mechanických závor jsem se rozhodoval mezi možnostmi kvalitnějšího - dražšího přestavníku, kdy bych místo jednoho serva, použil dvě (každé by bylo přímo pod závorami, a tak by se dal použít ocelový pevný drát jako neohebné pevné táhlo) a levnějšího přestavníku. Zvolil jsem levnější variantu a vyskytly se obtíže s měkkými táhly. Po několika laděních se nakonec podařilo vyrobit dostatečně fungující přestavník závor. Servo je umístěno na středu mezi závorami a každá závora funguje se dvěma táhly, která proti sobě působí opačným smyslem.

Ve výsledku jsou servo přestavníky dražší než přestavníky s Memwire, ale konstrukce je snazší. Servo je odolnější proti změnám teplot, je rychlejší, a přestavník se servem zabírá méně místa než Memwire. Pravdou je, že servo dělá poněkud větší hluk než Memwire.

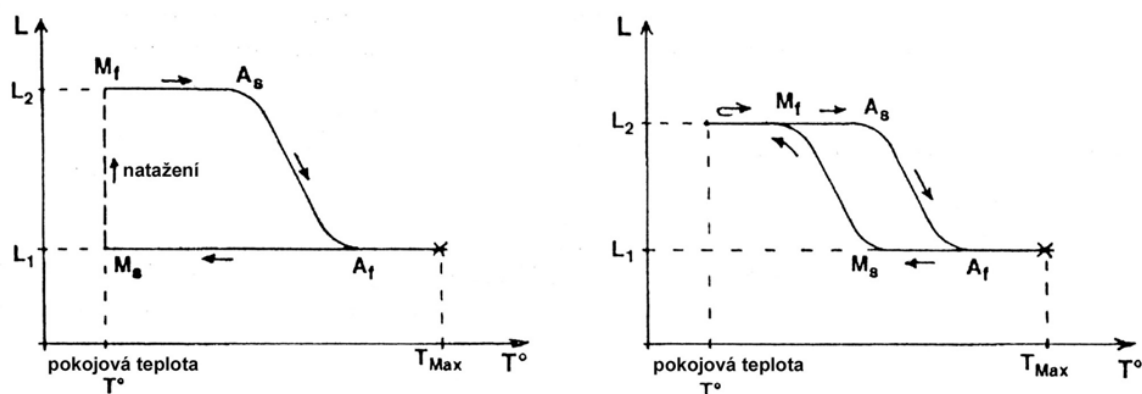
Skutečnou překážkou však pro mě byla programovací DCC kolej. Na všech webech jsem nacházel odkazy na DCC centrály, vyrobenými různými firmami, a jejich snadné nastavování CV registrů. Teoreticky bych pro další postup navrhoval použití sériové komunikace z PC do desky Arduina, vhodným způsobem poslat data do mikroprocesoru a ten z nich udělá DCC signál pro programovací kolej.

Použitá literatura

- [1] *Arduino* [online]. 2005, 2013 [cit. 2013-06-07]. Dostupné z: <http://www.arduino.cc/>
- [2] BARTOŠ, Václav. *Elektrické stroje*. Vyd. 1. Plzeň: Západočeská univerzita, 1995. ISBN 80-7082-221-x.
- [3] LITOMYSKÝ, Petr. *Litomyský.cz* [online]. 2001 [cit. 2013-06-02]. Dostupné z: <http://www.litomysky.cz/>
- [4] Martenzitická transformace. *Fyzikální ústav AV ČR, v.v.i.: Oddělení funkčních materiálů* [online]. 2013 [cit. 2013-06-02]. Dostupné z: http://ofm.fzu.cz/sma/brana_cz/info/mtransformace.htm
- [5] PINKER, Jiří. *Mikroprocesory a mikropočítače*. 1. vyd. Praha: BEN - technická literatura, 2004, 159 s. ISBN 80-730-0110-1.
- [6] PINTA, Martin. *Lokopin* [online]. 2006, 2013 [cit. 2013-06-07]. Dostupné z: <http://lokopin.wz.cz/body.htm>
- [7] Škola - Všeobecně o Servo Technice. *Pohonnatechnika.cz* [online]. 2007, 2013 [cit. 2013-06-03]. Dostupné z: <http://www.pohonnatechnika.cz/skola/servo/skola-vseobecne-o-servo-technice>

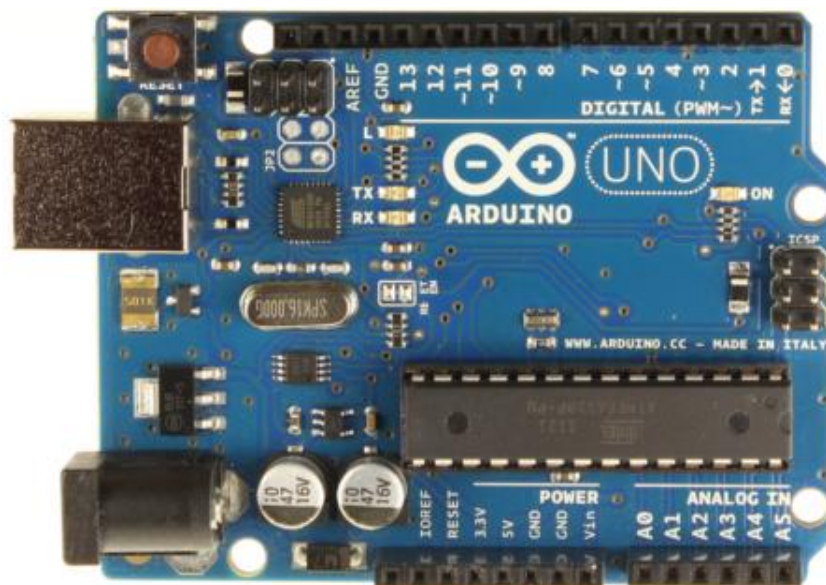
Přílohy

Příloha A



Jednosměrná (vlevo) a vratná (vpravo) tvarová paměť

Příloha B



Vývojová deska Arduino UNO

Příloha C

```
#include <Servo.h> // vložení knihovny Servo

Servo servoVyhybka; // Vytvoří objekt servo k ovladání serva Vyhybka a Zavor
Servo servoZavor; // Je možné vytvořit až 8 servo objektů

const int prepinacV = 2; // Přiradí prepinac vyhybky na pin 2 a prepinac zavor na pin 8
const int prepinacZ = 8; // Prepinac simuluje ovladání centralou (+5V na digitální vstup)

int posZ = 10; // proměnná, která uchovává pozici serva (vychozí pozice 10°)
boolean stavPrepinaceV; // proměnná, která uchovává stav prepinace Vyhybky
```

```

boolean stavPrepinaceZ;      // promenna, která uchovava stav prepinace Zavor

void setup()                 // cast programu, která se provede jen jednou
{
  servoVyhybka.attach(3);    // attach pripoji servoVyhybka na pin 3
  servoZavory.attach(9);     // attach pripoji servoZavory na pin 9

  servoVyhybka.write(0);     // write zapise do serva pocatecni pozici v nule
  servoZavory.write(posZ);   // write zapise do serva pozici ulozenou v promenne posZ

  pinMode(stavPrepinaceV, INPUT); // nastaveni pinu prepinaceV jako vstup
  pinMode(stavPrepinaceZ, INPUT); // nastaveni pinu prepinaceZ jako vstup
}

void loop ()                 // cast programu - loop (smyčka), která funguje porad dokola
{
  stavPrepinaceV = digitalRead(prepinacV); // digitalRead cte stav promenne prepinacV
  stavPrepinaceZ = digitalRead(prepinacZ); // digitalRead cte stav promenne prepinacZ

  if (stavPrepinaceV) {     // pokud je stavPrepinace HIGH (log. "1")
    servoVyhybka.write(45); // write zapise pozici 45° (odzkousena pozice vyhybky)
  }
  if (!stavPrepinaceV) {   // pokud není stavPrepinace HIGH (je LOW - log. "0")
    servoVyhybka.write(0);
  }

  if (stavPrepinaceZ) {    // pokud je stavPrepinace HIGH (log. "1")
    for(posZ;posZ<60;posZ++){ // zaveden cyklus for kvuli zpomalení otacek serva
      servoZavory.write(posZ);
      delay(15);           // delay nastavuje zpoždění v [ms] -> otocení serva o 1° za 15ms
    }
  }
  if (!stavPrepinaceZ) {  // pokud není stavPrepinace HIGH (je LOW -> log. "0")
    for(posZ;posZ>10;posZ--){
      servoZavory.write(posZ);
      delay(15);
    }
  }
}

```