

**ZÁPADOČESKÁ UNIVERZITA V PLZNI  
FAKULTA ELEKTROTECHNICKÁ**

**Katedra technologií a měření**

# **DIPLOMOVÁ PRÁCE**

**Inovace lineárního posuvu fotoaparátu**

**Vít Krejčí**

**Plzeň 2018**

**ZADÁNÍ DIPLOMOVÉ PRÁCE**  
(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Vít KREJČÍ**  
Osobní číslo: **E15N0020P**  
Studijní program: **N2612 Elektrotechnika a informatika**  
Studijní obor: **Komerční elektrotechnika**  
Název tématu: **Inovace lineárního posuvu fotoaparátu**  
Zadávací katedra: **Katedra technologií a měření**

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


1. Popište stávající stav lineárního posuvu fotoaparátu a definujte požadované inovace.
2. Navrhněte inovovaný lineární posuv fotoaparátu.
3. Podle návrhu zkonstruujte lineární posuv fotoaparátu.
4. Experimentálně ověřte funkčnost sestaveného lineární posuv fotoaparátu.

Rozsah grafických prací: **podle doporučení vedoucího**  
Rozsah kvalifikační práce: **40 - 60 stran**  
Forma zpracování diplomové práce: **tištěná/elektronická**  
Seznam odborné literatury:


**1. Určí vedoucí práce.**

Vedoucí diplomové práce: **Doc. Ing. Pavel Drábek, Ph.D.**  
Katedra elektromechaniky a výkonové elektroniky

Datum zadání diplomové práce: **16. října 2016**  
Termín odevzdání diplomové práce: **24. května 2018**

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



  
Doc. Ing. Aleš Hamáček, Ph.D.  
vedoucí katedry

V Plzni dne 10. října 2017

## **Abstrakt**

Práce se odkazuje na první typ konstrukce Lineárního kamerového posuvu, jenž je v úvodu zhodnocen, navrhuje jednotlivá zlepšení, úpravy a uvádí patřičné důvody. Z výsledných parametrů nabízí data pro konstrukci/výběr Lineárního posuvu s několika stěžejními body. Motorizace zvolené konstrukce se neobejde bez zásahu - modifikace. Implementace nových dílů klade minimální dopad na originální sestavení a zachovává jednoduchost. Finální konstrukce je osazena pohony posuvu a rotace. Navržena a zrealizována řídicí elektronika s krycími pouzdry. Řídicí program pohybu, uživatelského rozhraní a synchronizace fotoaparátu potřebné ke snímání časosběrů.

## **Klíčová slova**

Lineární posuv, camera slider, karbonové kolejničky, trubičkové kolejničky, 3d tisk, 3d modely, řemenice, krokový motor, domeček motoru, synchronizace fotoaparátu, časosběr, driver krokového motoru, grafický displej, bluetooth modul, ATmega128, ST7920, DRV8825, TB6560, HC-05.

## **Abstract**

The master thesis refers to the first type of the Linear camera slider. In the introduction is an evaluation of the previous construction with points describing individual improvements, modifications and several practical reasons for the change. The result of the gained rating parameters is a new construction with different goals. Secondary part is the motorizations of the selected construction. It is focused on implementation of the new mechanical parts with minimal impact to the original construction. In the final structure, there are applied two units of traction movement and rotation. The control electronic is designed and manufactured with a protective case included controls elements. The software components run the motors, show the user interface and manage the synchronization camera trigger. All these controlling activities are necessary for making time-lapse.

## **Key words**

Linear camera slider, carbon rails, linear move, tube rails, 3d print, 3d models, stepper motor, belt pulley, motor base, camera synchronisation, time-lapse, graphic unit, driver of stepper motor, bluetooth modul, ATmega128, ST7920, DRV8825, TB6560, HC-05.

## **Prohlášení**

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

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

.....  
podpis

V Plzni dne 24.5.2018

Vít Krejčí

## **Poděkování**

Velice rád bych poděkoval Doc. Ing. Pavlu Drábkovi, Ph.D za připomínky k mechanické konstrukci lineárního posuvu, vyřízení technických náležitostí plynoucí z motorové modifikace spojené s instalací nových dílů a sjednání kontaktů s dalšími odborníky. Zejména pak Ing. Bedřichu Bednářovi za technickou konzultaci 3D modelů, nápomoci při jejich realizaci a tisku. Panu Petru Kunstmüllerovi za zhotovení desky plošného spoje. Ing. Janu Pokornému za konzultaci zapojení mikrokontroléru k využití alternativních funkcí i následného softwarového zpracování dat. Otcí Robertu Krejčímu za praktické rady během úprav konstrukce a asistenci při montáži.

# Obsah

<b>ÚVOD</b> .....	<b>8</b>
<b>1 LINEÁRNÍ POSUV V1.0</b> .....	<b>9</b>
1.1 KONSTRUKCE .....	9
1.1.1 Délka pojezdu.....	10
1.1.2 Upevňovací body.....	10
1.1.3 Ložiska .....	11
1.1.4 Motor .....	12
1.1.5 Vozík.....	13
1.1.6 Řemen.....	14
1.1.7 Elektronika – umístění .....	15
1.1.8 Baterie.....	16
1.2 KONSTRUKČNÍ SOUHRN NEDOSTATKŮ .....	16
<b>2 INOVOVANÝ LINEÁRNÍ POSUV V2.0</b> .....	<b>17</b>
2.1 3D TISK.....	17
2.1.1 Princip, Technologie.....	18
2.1.2 Modelování .....	22
2.2 ASHANKS LINEAR SLIDER .....	25
2.2.1 Karbonová kolejnice .....	25
2.2.2 Vozík.....	27
2.3 ZMĚNY KONSTRUKCE .....	28
2.3.1 Vodičí tyč .....	29
2.3.2 Přestavba vozíku .....	29
2.3.3 Pohon vozíku.....	31
2.3.4 Krabička elektroniky.....	33
2.3.5 Volnoběžná kladka .....	33
2.3.6 Baterie.....	34
2.4 ZHODNOCENÍ KONSTRUKCE .....	35
2.4.1 Vodičové spojení .....	35
<b>3 ELEKTRONIKA ŘÍZENÍ – JEDNOTKY POHONU</b> .....	<b>36</b>
3.1 NÁVRH ŘÍDICÍ ELEKTRONIKY .....	36
3.1.1 Realizace .....	37
3.1.2 Výroba plošného spoje .....	37
3.2 TYPY MODULŮ.....	38
3.2.1 Napájecí měnič.....	39
3.2.2 Mikrokontrolér.....	40
3.2.3 Silové spínací obvody krokového motoru.....	40
3.2.4 Bluetooth modul HC- 05 .....	43
3.3 POHON .....	44
3.3.1 Manipulátor vozíku .....	44
3.3.2 Manipulátor těla fotoaparátu.....	47
3.3.3 Zpětná vazba .....	47
3.4 ZDROJ NAPÁJENÍ .....	50
<b>4 PROGRAMOVÁNÍ</b> .....	<b>51</b>
4.1 OŽIVENÍ.....	52
4.1.1 Fuse bity.....	52
4.1.2 Nastavení bran .....	53
4.1.3 Nastavení hodin .....	53



---

4.1.4	<i>Vnější přerušení</i> .....	55
4.1.5	<i>Grafický displej</i> .....	55
4.2	FUNKCE PROGRAMU .....	56
4.2.1	<i>Foto režim</i> .....	56
4.2.2	<i>Video režim</i> .....	57
4.2.3	<i>Bluetooth režim</i> .....	58
4.3	KONFIGURACE MOTORŮ .....	58
4.4	STRUKTURA PROGRAMU .....	59
4.5	OVLÁDÁNÍ ZAŘÍZENÍ .....	59
<b>5</b>	<b>CENOVÁ KALKULACE</b> .....	<b>60</b>
	<b>ZÁVĚR</b> .....	<b>61</b>
	<b>SEZNAM LITERATURY</b> .....	<b>64</b>
	<b>PŘÍLOHY</b> .....	<b>1</b>

## Seznam symbolů a zkratk

$m$	Hmotnost [kg]
$g$	Gravitační zrychlení [ $m/s^2$ ]
$F$	Síla [N]
$r$	Poloměr řemenice motoru [cm]
$M$	Moment motoru (s řemenicí) [Nm]
$V_{in}$	Vstupní napětí [V]
$L_{min}$	Minimální indukčnost [H]
$V_{sat}$	Saturační napětí [V]
$V_{out}$	Výstupní napětí [V]
$I_{pk}$	Špičkový proud [A]
$t_{on}$	Perioda spínání měniče [s]
$i_{vin}$	Okamžitá hodnota proudu vinutím [A]
$U_{mot}$	Napětí na svorkách motoru [V]
$R_{vin}$	Odpor vinutí [ $\Omega$ ]
$u_L$	Okamžitá hodnota napětí cívky [V]
$\lambda$	Převrácená časová konstanta vinutí [ $s^{-1}$ ]
$\tau$	Časová konstanta vinutí [s]
$Bd$	Přenosová rychlost [ $\frac{bit}{s}$ ]
$f$	Frekvence [Hz]
$T$	Perioda [s]
$\omega$	Úhlová rychlost [ot/s]
$v$	Rychlost [mm/s]
$s$	Dráha [mm]
$t_{pře.}$	Čas přejezdu [s]
$O$	Obvod [mm]

## Zkratky

FFF	Fused filament fabrication	DIR	Direction (směr otáčení hřídele)
ABS	Akrylonitrilbutadienstyren	STP	Step (krok pootočení rotoru)
PLA	Polylactic acid	ENB	Enable (povolení činnosti)
PVA	Polyvinylalkohol	MCU	Micro Controller Unit
CAD	Computer aided design	SPI	Serial Peripheral Interface
DIY	Do it yourself	ISP	In System Programming
3D	Three dimensional	PWM	Pulse Width Modulation
STL	Standard Template Library	DPS	Deska plošného spoje
CNC	Computer Numerical Control	GND	Ground
SD	Secure Digital card	TTL	Transistor transistor logic
GCODE	Souborové označení kódu		

## Úvod

Práce navazuje na již započatý projekt z bakalářského studia. Pojednává o konstrukci fyzického zařízení, lineárního posuvu pro fotografické/filmové účely. U nově vznikající konstrukce se zaměříme na drobná vylepšení plynoucí z praktického užívání zařízení.

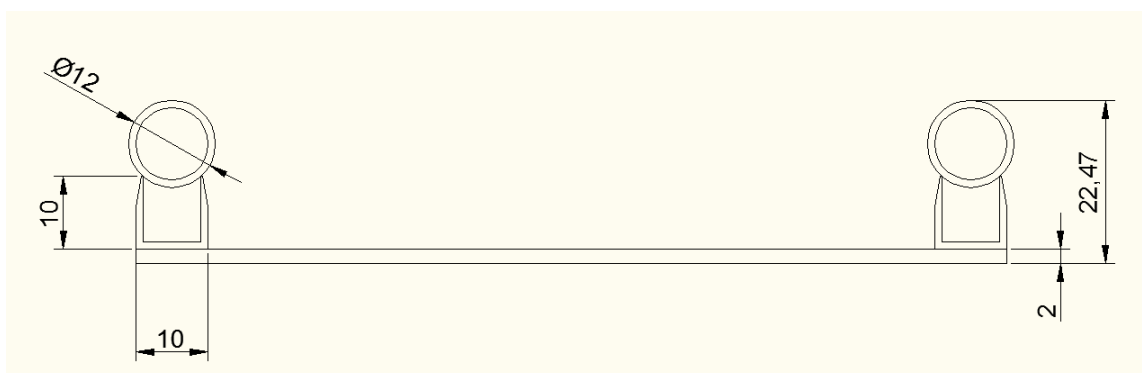
Text vysvětluje nově zvolený typ konstrukce s příslušnými transformacemi, změnu pohonu s důrazem na větší točivý moment. Rozvržení elektroniky pro řízení pohonu vozíku doplněnou o funkci rotace těla fotoaparátu. Poslední část se věnuje softwaru pro ovládání elektroniky a vzdálenému řízení lineárního slideru.

## 1 Lineární posuv v1.0

První verze lineárního posuvu začala vznikat v druhé polovině roku 2015. Koncept byl založen na vybudování vlastní jednoduché konstrukce, tak aby ji bylo možné zrealizovat i v „domácích“ podmínkách. Většina náročnějších technických dílů byla převzata z vyřazených zařízení.

Řídící elektronika lineárního posuvu je osazena procesorem ATmega32 zajišťující ovládání pohonné jednotky vozíku, přesněji krokového motoru. Řízený posun vozíku je synchronizován se spuštěním fotoaparátu. Dochází k cílenému zastavení snímací soupravy a opětovnému uvedení do chodu. Sousednost tohoto manipulačního řízení je specifická k sekvenčnímu snímání nebo-li časosběrnému fotografování.

Nynější práce bude čerpat z těchto nabytých zkušeností a zohlední zde praktické poznatky vzniklé při užívání lineárního posuvu v exteriéru.



Obr. 1 - Vizuální podoba prvního projektu (řez kolejiště).

### 1.1 Konstrukce

První typ konstrukce vynikal svou dostupností, jež byla sestavena z běžně nabízených hliníkových profilů. Montáž kolejí s nabízeným technickým zázemím nezaručovala dodržení přesné rozteče kolejí. Nicméně, dovolovala sestavit až dvoumetrovou kolejnici s požadovaným rozvorem. Využitím několika tvarů profilů a spojek byla konstrukce dostatečně pevná. Spoje byly fixovány vruty, avšak před dotažením muselo být vše pečlivě centrováno.

Z tohoto důvodu nemohla být konstrukce jednoduše rozebrána a složena jako stavebnice (transportní obtíže). Ideální konstrukce by umožňovala rozložení do čtyř dílcích celků:

- *Vodící kolejnice*
- *Příčné spojníky*
- *Vozík*
- *Krabička s elektronikou a pohonnou jednotkou*

### 1.1.1 Délka pojezdu

Délka kolejiště se stanovila na rozměr 155cm a manipulační délka pojezdu vozíku je v rozsahu 130cm. S ovládací elektronikou se dostáváme na finální rozměr zařízení 172cm.

Takto rozměrná konstrukce se v praktickém používání, zejména při manuálním přenosu na odlehlá místa neosvědčila a délka posuvu byla značně předimenzována. Cílem je tedy konstrukci zkrátit a celý koncept zpracovat kompaktněji s důrazem na mobilitu při ztrátě delší pojezdové vzdálenosti.

### 1.1.2 Upevňovací body

Lineární posuv používáme ve dvou režimech:

- *Snímání ze stativu*

K uchycení stativu slouží otvory ve spojových plátech kolejnic. S jedním fotografickým stativem není však možné konstrukci vynést a jsou potřeba dva opěrné body.

Toto je jedna z dalších příčin nevhodná pro práci v exteriéru v důsledku náročnosti množstevního vybavení k transportu. Náročnost se rovněž pojí s nastavením vhodné výšky stativu a optimálního horizontálního sklonu vozíku.

- *Snímání z pevného podloží*

Vyhovuje-li nám výška podloží, případně snímáme objekt ze země, využijeme „nožiček“ které podírají konstrukci několik centimetrů nad zemí. Nožičkové stojky by měli být obvykle nastavitelné, aby mohlo opět dojít k mírnému vyrovnání horizontální roviny.

Nožičky, ačkoliv nebyly plánovány, se do konstrukce přidaly až v samotném zakončení celého výrobku. K montáži došlo v důsledku „domečku“ držící krokový motor viz Obr. 5 - Přesah motoru pod úroveň konstrukce. Spodní část přesahuje přes hranu kolejnice a způsobuje nedoléhání kolejnic k zemi.



Obr. 2 - Regulovatelné podpěrné nožičky.

### 1.1.3 Ložiska

Další nedostatek se objevil u použitých lineárních ložisek s uzavřenou kuličkovou dráhou v pouzdře. Obecně tyto typy vedení jsou vhodná pro průmyslové použití a pevné konstrukce vyžadující relativně čistý povrch pokryt tenkou vrstvou tuku. V tomto případě pracují velmi spolehlivě a přesně. Pokud je ale posuv využíván v externích podmínkách za přítomnosti deště, prachu či zrnkových částic písku se kolejnice ošetřena vrstvičkou mazacího tuku stávají spolehlivým lapačem nečistot. V důsledku těchto okolností a podmínek se použití průmyslově přesného vedení nejeví výhodné. Nepříjemnou skutečností je i nechtěný styk dlaní s mazacím tukem, ke kterému dochází při manipulaci a přenosu konstrukce.

#### Shrnutí v bodech:

- Pravidelná údržba vedení (mazání a čištění)
- Zadržování zrnkových částic písku
- Hmotnost
- Mazací tuk kolejnic (manipulace)



Obr. 3 – Stírání nečistot z kolejí.

K vyloučení problému s nečistotami je zapotřebí použít „suchého“ vedení. Tomuto způsobu vedení nejlépe odpovídají valivé systémy se zapouzdřenými ložisky. Jedná se o kuželovou vodící kladku tvořenou z pryže se zapouzdřeným ložiskem v ose. Systém na kolejnicích perfektně sedí, tření je zanedbatelné a údržba minimální.

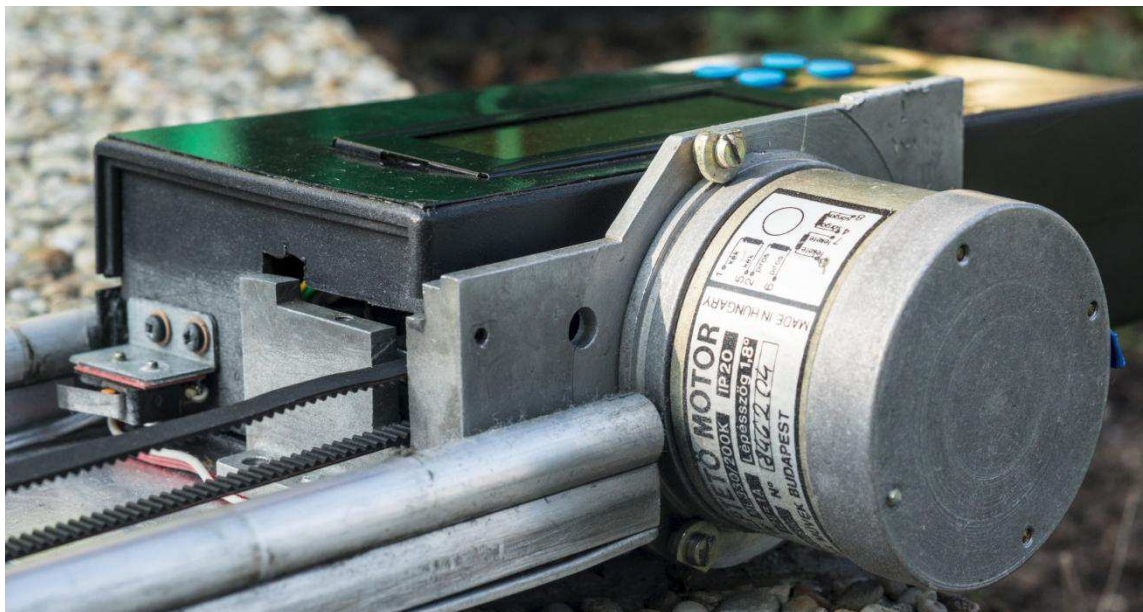
#### 1.1.4 Motor

Pohonná jednotka tvořená krokovým motorem splnila požadavky pro posuv jen v horizontální rovině. Pro větší náklon či vertikální posuv je vyvíjený moment nedostatečný. Motor je rovněž náchylný na pulzování a rozvibrování celé konstrukce při určitém pásmu spínacích frekvencí.

Eliminaci problému vibrací řešíme tedy úpravou konstrukce (vložením pryže do míst šroubových spojů) či vhodnou konfigurací softwaru řídící krokování rotoru.

K dosažení maximální efektivity motoru, především v rozsahu vyšších otáček je nutné zaručit dostatečný příkon k nabuzení cívek statoru. Proto bude nutné se zaměřit na napájecí a řídicí části krokového motoru.





Obr. 4 - Uchycení motoru mimo osu kolejiště.

- **Těžiště**

Hybridní krokový motor je malý stroj o poměrně velké hmotnosti (s točivým momentem potřebným pro kamerovou techniku) přibližující se k hranici 600g. Umístění domečku pro pohonnou jednotku je tedy vhodné provést tak, aby těžiště motoru bylo v ose kolejiště konstrukce a nedocházelo k mechanickému pnutí zejména na krut.

Popisované rozpoložení opět zlepší manipulační vlastnosti jakožto konstrukčního celku.



Obr. 5 - Přesah motoru pod úroveň konstrukce, nutno použít podpěrné nožičky.

### 1.1.5 Vozík

Základna vozíku je zkonstruována z hliníkového plechu. Podstavu tvoří ložiska ukotvená na třech bodech v rozích desky. Ve spodní části jsou umístěny klemy pro sevření řemene.



Stativový šroub umístěn ve středu vozíku využíváme k připevnění samotného fotoaparátu nebo stativové hlavy, jenž nám dovolí lépe kompenzovat náklon a vyrovnání horizontální roviny pro ideální kompozici záběru.

Další možnosti požití nejsou momentálně k dispozici.



Obr. 6 - Spodní strana vozíku, řemenové svěrky, stativový šroub.

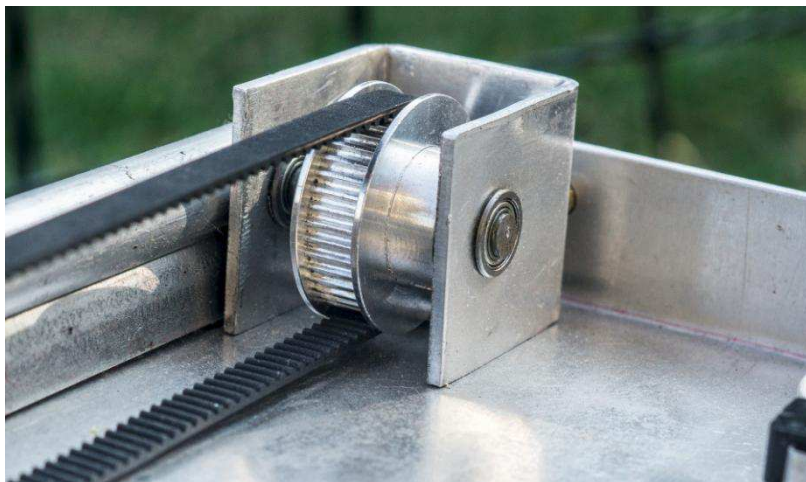
### 1.1.6 Řemen

Prostor řemenové smyčky je pevně definován postavením pohonu (příp. volnoběžné řemenice).

- *Svislá smyčka*

Řemen obíhá pojezdovou dráhu ve dvou výškových úrovních v těsné blízkosti jedné z kolejí, přičemž horní část řemene je sevřena k vozíku.

V tomto rozpoložení nehrozí smeknutí řemene mimo dráhu řemeničky vlivem gravitační síly (zejména u delších posuvů = delší řemen). Osa motoru je kolmo ke kolejišti, v případě umístění větších motorů je problematické zachovat motor v těžišti.



Obr. 7 - Řemenička, svislá smyčka.

- **Plochá smyčka**

Smyčka kopíruje vnitřní obvod kolejí. Osa motoru je umístěna vždy ve středu kolejiště. Jednodušší snížení motoru pod úroveň vozíku z důvodu kolmosti osy na podstavu kolejiště (není nutno řešit přesah těla motoru pod úroveň kolejiště).

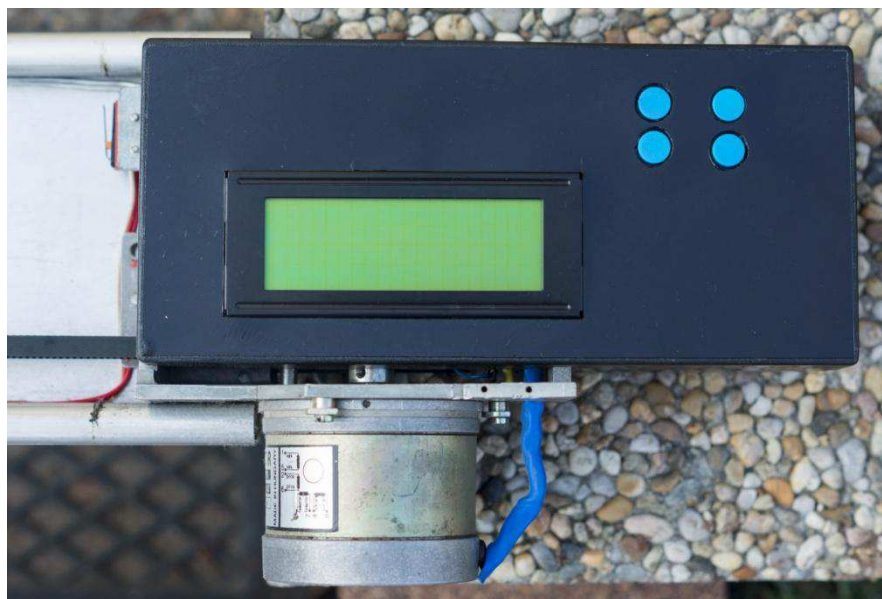
### 1.1.7 Elektronika – umístění

Řídící elektronika je umístěna v blízkosti motoru na jednom z konců lineární dráhy. Přenosovou cestu od výkonových spínačů k motoru se snažíme co nejvíce zkrátit z důvodů parazitních vlivů na vedení.

- **Pouzdro**

Elektroniku je nutné krýt před vnějšími nepříznivými vlivy počasí a rovněž zajistit mechanickou odolnost. Krabička svým tvarem ideálně zůstává schována v konstrukci slideru, případně navazuje a zakončuje jednu stranu lineárního vedení.

Plošný spoj musí být již navrhován do vnitřních prostor krabičky se všemi nezbytnostmi (např.: chlazením, konektorovými vývody, šroubových bodů, umístění displeje, atd.).



Obr. 8 - Řídící elektronika, rozmístění ovládacích prvků.

### 1.1.8 Baterie

Zdrojem elektrické energie je olověný akumulátor s nominálním napětím 12V a kapacitou 4 Ah. Spotřebu zařízení spolehlivě vykryje. Nicméně hmotnost zdroje není adekvátní k účelům manipulace a častému transportu.

Řešení je v již cenově dostupných Li-Ion článcích typu 18650. Články poskládáme vhodnou sérioparalelní kombinací k dosažení optimálního napětí pro efektivní provoz motorů. Sestavenou baterii lze použít jako protizávaží k motorové jednotce pohonu vozíku.

### 1.2 Konstrukční souhrn nedostatků

- *Podpěrné nožičky* - *nestabilita*
- *Rozměrnost* - *mobilita*
- *Umístění motoru* - *osa těžiště, přesah pod konstrukci*
- *Pojezdová ložiska* - *hmotnost, údržba*
- *Jednodílná konstrukce* - *nelze rozložit pro transport*

## 2 Inovovaný lineární posuv v2.0

V samotném počátku musím zhodnotit limity, kterých jsem dosáhl se stavbou první konstrukce. Nepřesné šroubové spojení profilů v délce kolejového vedení má za následek nekonzistentní pohyb vozíku. Kvalita a opracování materiálu není strojově přesná. Hmotnost komponentů je přehlížena, zabýváme se pouze funkčním výsledkem.

Proto konstrukční část s přihlédnutím k ekonomické a časové stránce nebylo možné efektivně zlepšit i za podmínek dostatečně vybavené pracovní dílny. Rozhodnutím bylo poohlédnout se na trhu komerčně dostupných sliderů. Kritéria konstrukce vycházela z nově nabytých praktických zkušeností s používáním posuvu, tak jak byla zhodnocena v první kapitole.

Tab. 1 – Hledané parametry.

nízká hmotnost	možnost úpravy pro motorizaci
kompaktnost	jednoduchost
délka posuvu do 1m	středová destička pro upnutí stativu
zajištění vozíku pro vertikální zdvih	integrované vyrovnávací nožičky

Jednalo se tedy o konstrukce bez pohonné jednotky. Vhodným výběrem se jistě ušetříme nejsložitějším komplikacím, které by byly konstrukčně náročné, nicméně mechanickým pracím se nevyhneme. K upevnění částí související s pohonem nebo elektronikou bude nezbytné do zařízení doplnit. Tuto činnost by nám měl nejvíce usnadnit 3d tisk.

### 2.1 3D tisk

Vzniklé obtíže, související s vymýšlením a instalací jednoúčelových vylepšení (modifikací) na zařízení v dostatečně přesném provedení se dnes nejvíce osvědčuje 3d tisk. Součástka/díl je přesně přizpůsobena zařízení, kopíruje jeho linie, využívá montážních otvorů a jeho mechanické vlastnosti lze volit dle použitého materiálu a zvolené struktury tisku. Tyto vlastnosti umožňují vytisknutému dílu plně splynout se zařízením, do kterého byly vloženy a

to s minimálními zásahy do původní konstrukce. Operace provedené na zařízení jsou v tomto případě vratné a po demontáži dostáváme opět původní zařízení bez poškození.

### 2.1.1 Princip, Technologie

3D tisk je oproti ostatním známým způsobům tvorby výrobků tzv. aditivním. Forma vytváření třídimenziálního objektu je založena na přidávání materiálu. Naopak většina průmyslových strojů např.: soustruh a fréza materiál ubírají. Aditivní technologie využívá několik způsobů tvorby objektů související se zpracováním – úpravou povrchu, druhem materiálu, kusové/množstevní výroby, atd. [4 -5]

Momentálně nejdostupnější a nejrozšířenější jsou tiskárny Reprap s tryskovou hlavou tavící vlákno ze zásobníkové cívky. Typ trysky určuje průřez taženého vlákna související s přesností, detailem a rovněž i časovou náročností tisku. Vlákno je kladeno na temperovanou podložku v horizontálních vrstvách. Vyhřívání zajišťuje konstantní teplotu modelu po dobu tisku. V opačném případě by se materiál vlivem rychlého ochlazení zdeformoval. Po vyplnění plochy protínající řez tělesa se tisková hlava posune o úroveň výš a pokračuje další vrstvou.

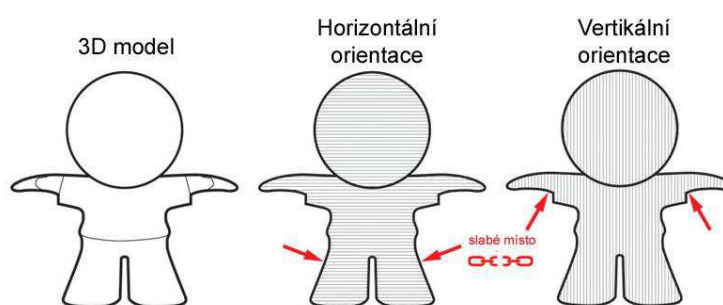
Tento způsob tisku je označován jako FFF (fused filament fabrication). Tento princip kladení vlákna nese i několik dalších pojmenování. Obecně fungují na stejné bázi [4].

- ***Způsob tisku***

Pro tisk třídimenzionálního objektu vyžadujeme tři pohyblivé osy, s kterými tiskárna pracuje. Nahlížíme-li na tiskárnu z půdorysu, můžeme říci, že tisková hlava se pohybuje v osách X a Z, třetí osový pohyb Y vykonává podložka pro model. Tisknutí vrstvy (řezu) modelu probíhá za pomoci koordinátorů X a Y, dojde-li k vykreslení celé vrstvy, hlava se posune po ose Z výš do následující nové řezové hladiny. Minimální krok posuvů os určuje výsledné rozlišení – přesnost. Obvykle se pohybují v jednotkách desetin milimetru.

Proces tisku je zahájen navíjením vlákna z vybrané cívky do tiskové hlavy, kde je vlákno roztaveno. Mezitím tiskárna provede kalibraci os a nahřeje podložku pro model. Jsou-li kroky splněny, začne tisk modelu.

Připravený model před samotným tiskem je zapotřebí vhodně umístit vzhledem k podložce a směru tažení vlákna. Ke způsobu postavení modelu se budou odvíjet určité mechanické vlastnosti. Podstava objektu má po většinou jinou strukturu povrchu, jednotlivá vlákna jsou teplotou podložky „natavena“ a povrch se slívá. Vytváří se tedy jiná struktura než zbytek povrchu<sup>1</sup>.



Obr. 9 – Směr tažení vlákna (převzato z [12]).

Druhé zohlednění se týká tenkých míst. Pokládání vlákna by mělo tenká (úzká) místa protínat ve směru větší mechanické zátěže. Riziko odtržení vlákna ve vrstvě je vyšší než narušení samotného vlákna tažené hmoty (spoj bude pružnější, ale nedojde k odlamování spojů mezi vláknami).

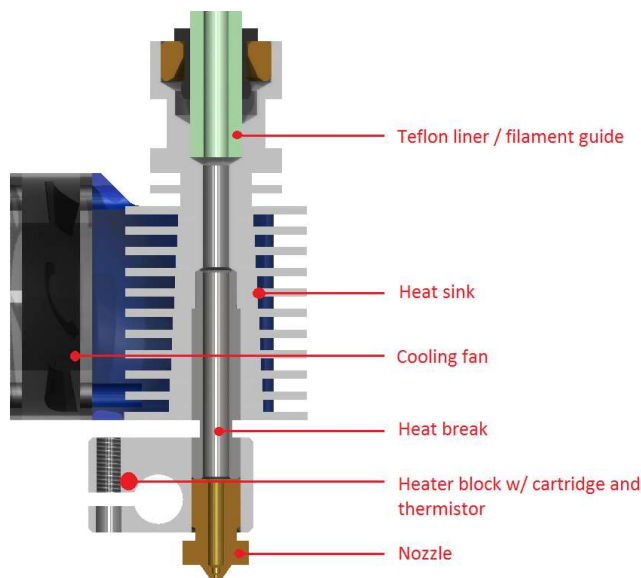
Možnost přemostění vlákna v prostoru (bez opory) je rovněž limitováno a spolehlivě přemostí jen několik centimetrů. Pro delší vzdálenosti dochází k průhybům a např.: rovná linka vlákna začíná až ve třetí, čtvrté vrstvě. Více informací k tisku modelu bude zmíněno v 2.1.2. modelování.

- **Tiskové hlavy**

Umožňují střídání materiálu či barvy během procesu tisku. Výsledný model tak může dosahovat lepších parametrů. Nosné části jsou tisknuty z pevnějšího ale hůře tvarovatelného materiálu. Zatímco zbytek je tisknut tvárnějším materiálem (působivější design) avšak s vyšší

<sup>1</sup> Na tuto vlastnost si dáváme pozor zejména u pohledových částí výrobku.

teplotní závislostí. Druhou možností je tisk podpůrných částí modelu z materiálu, jenž odstraní rozpouštědlo. Podpěrné struktury není proto nutno vylamovat a po zhotovení modelu se vloží jen do požadovaného roztoku. Využívá se u složitých modelů.[7 - 6]



Obr. 10 – Tisková hlava, jednotlivé části (převzato z [8]).

Změna barvy materiálu se uplatňuje zejména u textových popisů, vizuálních symbolů, značek, a atd.

- **Materiály**

Na základě tepelného zpracování vlákna v tiskové hlavě se jedná o tepelně závislé materiály, z nichž nejrozšířenější jsou termoplasty. Po zahřátí termoplastu, polymer postupně měkne a mezimolekulární interakce (van der Waalsovými silami) postupně slábnou. Teplota tání začíná zhruba na 100°C, dle konkrétního materiálu. [9 - 10]

**ABS** – (Akrylonitrilbutadienstyren) Pevný, mírně ohebný plast s vyšší teplotou tání. Používán převážně pro díly s vysokým mechanickým namáháním. Uplatní se i pro venkovní použití, odolá teplotám i UV záření. ABS má zakalený mléčný vzhled, nicméně je dostupné i v různých barvách. Rozpouští se v acetonu, které slouží rovněž jako lepidlo pro spojování dílů či úpravě (leštění povrchu). [10]



Tab. 2 – ABS teplotní parametry.

<b>Teplota podložky</b>	90 - 110 ° C
<b>Pracovní teplota tisku<sup>2</sup></b>	230 - 250 ° C

**PLA** – (Polylactid acid) Přírodní polymer vyráběn ze zemědělských plodin kukuřice, cukrové řepy, brambor, atd. Klade menší nároky na životní prostředí. Snadno rozložitelný, není vhodný pro venkovní použití. Mechanické vlastnosti (ohebnost) jsou horší než u ABS. Teplotní závislost je vyšší a může dojít ke změnám tvaru modelu již za podmínek dobře vyhřátého auta v letních dnech (začíná měknout již kolem 60 °C, zatímco ABS až od 100 °C). Povrch výrobku má lesklý vzhled. Při tisku dosahujeme větší přesnosti než u ABS, především kvůli pracovní teplotě, jenž souvisí s roztahováním/smršťováním materiálu během tisku/hotovou vrstvou. [9-10]

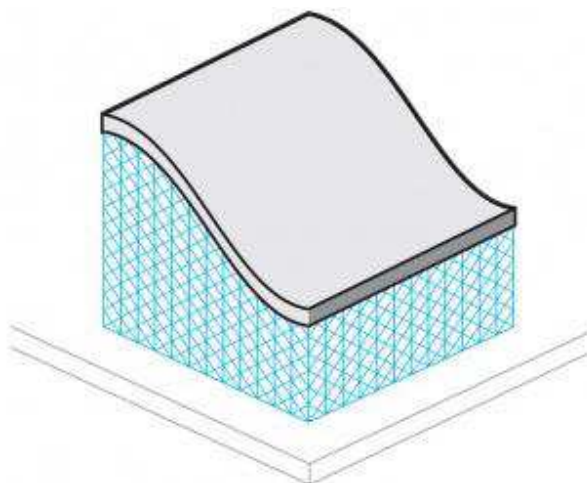
Tab. 3 - PVA teplotní parametry

<b>Teplota podložky</b>	0 – 60°C
<b>Pracovní teplota tisku</b>	185 – 235 °C

**PVA** – (Polyvinylalkohol) Vodou rozpustný syntetický polymer, určený primárně pro podpůrné konstrukce, výplně modelu. Jednoduše zpracovatelný a vrstvitelný. Pro odbourání konstrukčních prvků je model vložen do horké vody. [9]

Tab. 4 - PVA teplotní parametry

<b>Teplota podložky</b>	0 – 50°C
<b>Pracovní teplota tisku</b>	180 – 200 °C



Obr. 11 – Podpůrná konstrukce modelu (převzato z [12]).

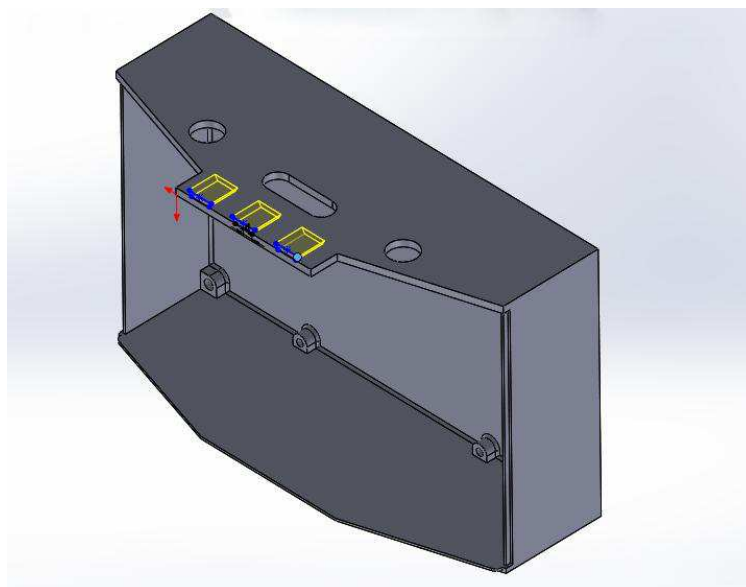
<sup>2</sup> Rozsah teplot je dán rychlostí tažení vlákna (filamentu) tryskou.



### 2.1.2 Modelování

Nabízejí se dvě možnosti. Jednodušší cestou je náhled do online databáze dílů, modelů poskytující webové servery. Nalezneme zde zpravidla díly pro vylepšení komerčně dostupných výrobků tzv. modifikace DIY<sup>3</sup>. Zbylou část databáze vyplňují povětšinou designové věci, figurky, atd. Vybraný model je možné si otevřít v jednoduchém webovém editoru, který umožňuje základní úpravy. Upravenou část lze uložit a exportovat pro tisk.

Důmyslnějším řešením je využití sofistikovaného CAD softwaru pro realizaci 3d objektů. Díl navrhujeme přímo na míru dané aplikaci. Maximalizujeme efektivitu dílu k objemu materiálu. Realizujeme uzpůsobení montážních otvorů, vyztužení namáhaných částí, vkládáme pomocné úchyty, vzpěry, žebra atd. Složité modely je třeba rozdělit na více částí, tak aby byly realizovatelné na 3d tiskárně. Tisk bude probíhat totožně jako u vícedílných modelů s tím rozdílem, že vícedílné modely spojujeme rozebíratelně (šroubově) např.: krabička, ale složité modely obvykle spojujeme lepením (nerozebíratelný spoj) nebo určitým druhem mechanického zámku např. rybina. Omezení vztahující se k připravovanému modelu mohou být rozměrová, závisující na velikosti tiskové podložky a tvarová odvíjející se od složitosti dílu s důrazem na eliminaci podpůrné konstrukce resp. odpadního materiálu. Díl rovněž vyžaduje rovnou plochu tělesa, sloužící jako podstava pro zbytek hmoty jednotlivých vrstev.

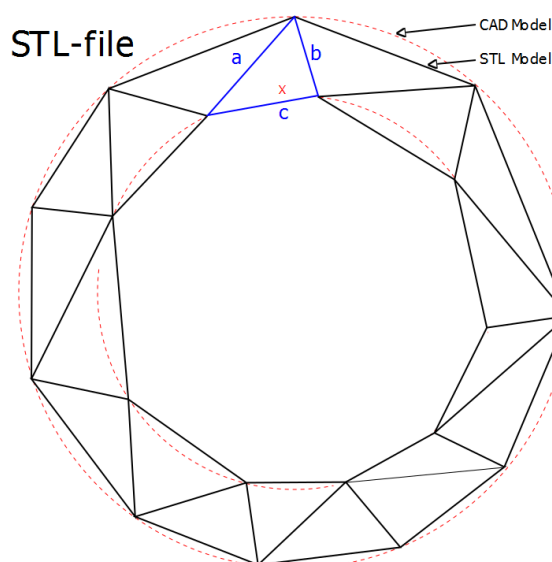


Obr. 12 – Vytváření mechanického zámku, rybiny, označena žlutě.

<sup>3</sup> Do it your self, přeloženo jako udělej si sám.

V modelu je zapotřebí navrhnut všechny potřebné otvory, pozdější vrtání do těla objektu může narušit pevnost, především otvor pro šroub nebude pevný. Vnitřní objem tělesa není vyplněn hmotou nýbrž strukturou (materiál + vzduch). Struktury je využíváno pro úsporu vlákna (materiálu), snížení hmotnosti i zkrácení času tisku. V rámci navýšení pevnosti se doporučuje zaoblovat veškeré hrany a velké plochy tělesa vyplnit libovolným vzorem otvorů, drážek přispívající ke zpevnění struktury<sup>4</sup>. Členitý profil na povrchu výrobku odolá lépe mechanickým vlivům.

Navržený díl exportujeme do souboru STL. STL z modelu odstraňuje redundantní informace o textuře, barvě atd. Ponechává informace pouze o bodech tělesa a charakterizující vazby mezi body propojeny do trojúhelníků. Koordináty jsou zaznamenány v kartézské soustavě (x, y). Soubor nezaznamenává jednotky, ale pouze měřítko, ve kterém bude objekt tisknut.[11, 13]



Obr. 13 – Řez modelu, princip vykreslení STL vrstvy (převzato z [13]).

Vygenerovaný STL formát používáme jako vstupní data pro software připravující konkrétní data pro vybranou 3d tiskárnu. Software může být založen na Opensource platformě, nebo je unikátní a dodáván společně s tiskárnou. Software umožňuje měnit desítky parametrů, které mají vliv na výsledné provedení modelu. Nastavení se odchyľuje dle účelu modelu. Zaměření pro konstrukční věci (mechanická pevnost), povrchový vzhled (figurky), úspora materiálu (objemné objekty). [12, 14]

<sup>4</sup> Tiskárna v daném místě položí více materiálu s větší hustotou vláken než u objemové struktury.

### **Výška vrstvy**

Parametr nastaven na 75% průměru trysky. Tak aby se vrstvy mezi sebou dostatečně spojily (překrývají se z 25% průměru taženého vlákna). S menším zdvihem trysky bude povrch jemnější, ovlivňuje finální vzhled výrobku.

### **Tloušťka stěny**

Uplatněný u konstrukčních dílů. Vzhled výrobku se nemění. Stěny modelu jsou vyplněny hmotou ve větší vrstvě (násobky).

### **Rychlost tažení plastu**

Upravuji-li výšku vrstvy, měním s ním i rychlost průchodu vlákna k trysce. Je-li zdvih trysky vyšší, průtok hmoty se musí zvýšit. Naopak při tisku figurek se průtok snižuje.

### **Teplota**

Vztahuje se vždy ke konkrétnímu materiálu viz tabulka 2 – 4. Nicméně k vybrání přesné hodnoty z rozmezí existují kalibrační modely, které tisknou úseky vrstev v postupně rostoucích teplotních škálách. Zhodnocením výtisku zvolíme optimální teplotu.

### **Rychlost tisku**

Opět má dopad na vzhled modelu. Vyšší rychlosti motoru mohou rozvibrovat konstrukci, hluk tisku narůstá.

### **Výplň – struktura**

Hodnota výplně je defaultně nastavena na 15 – 20% objemu hmoty. Strukturu (tvar) výplně můžeme definovat od libovolných 3d obrazců až po volitelné mřížky či včelí plástve.

### **Chlazení**

Definuje nastavení otáček ventilátoru chladící model. Otáčky korespondují s typem tiskového materiálu a velikostí modelu. Chlazením zrychlujeme vytvrzení výrobku, tak aby se nezhroutil pod vlastní vahou. Rovněž poslední vrstva musí zůstat tepelně schopná přilnout k vrstvě následující.

S vybraným nastavením spustíme proces generování jednotlivých vrstev řezů modelu, vypočítá se trasa tiskové hlavy, zobrazí se podpůrná konstrukce pod objekty v prostoru,

spotřeba materiálu a odhadovaný čas. Nově vzniklý soubor .GCODE vytvořený pro daný typ tiskárny zkopírujeme na SD kartu a vložíme do tiskárny. Z menu nabídky už jen spustíme tisk. [14]

## 2.2 ASHANKS linear slider

Popsanému zadání parametrově vyhověl lineární kamerový posuv Ashanks. Jedná se o produkt čínského výrobce zabývající se mechanickým vybavením stabilizátory, stativy, osvětlením pro kamerovou techniku nabízející zboží na populárním zahraničním e-shopu. S dostupností lineárního posuvu nebyly obtíže, cenová nabídka byla zajímavá a pokryla ze 2/3 náklady na hlavní komponent, karbonové trubičky již byly porovnávány vzhledem k cenám českým obchodům. Konstrukce mimo jiné obsahovala i další klíčové prvky, jenž by byly složitě realizovatelné i z hlediska použitého materiálu.

Produktu dominují karbonové kolejnice, plynulé vedení vozíku, koncové příčnický ze slitiny hliníku, možnost rotačního pohybu kamery na vozíku či různé další detaily pro zajištění (blokaci) posuvu i rotace. Rozpoložení konstrukce uspokojovalo podmínky pro motorovou přestavbu posuvu a řešilo zásadní problematické body.

Tab. 5 – Výčet parametrů.

Pojezdová dráha 100cm	Hmotnost 1,63 kg
Karbonová kolejnice	Integrované nožičky
Osové otáčení kamery	Středová upínací destička

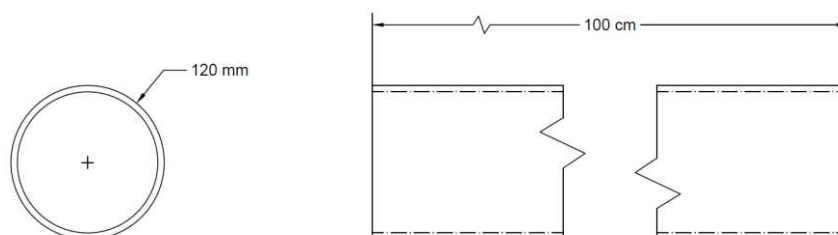
### 2.2.1 Karbonová kolejnice

Kolejnice tvoří dvojce karbonových trubiček excelující nízkou hmotností, vysokou pevností, dlouhou životností a bezpečně absorbují vnější rázy.

Pro zaručení soudržnosti celé konstrukce, kde kolejnice představují většinou část výrobku, volíme větší průměr profilu (trubička je samonosná a není podepřena další částí). Eliminujeme tak nežádoucí průhyb konstrukce, zvýšíme plynulost (odolnost koleček vozíku na drobné částečky na kolejišti) a v neposlední řadě získáme možnost lepšího úchopu kolejnice při manipulaci či využití vnitřního prostoru profilu k vedení kabeláže.

Vzhledem k snadné tvárnosti karbonového vzoru (tkaniny) před jeho vytvrzením pojivem je vhodnější vytvářet specifické profily s tenkou stěnou. Tím se dosahuje větší pevnosti, při menším množství použitého materiálu.

Profil je definován vnějším průměrem 120mm a tloušťkou stěny 1mm. Délka profilu pak dosahuje k jednomu metru.



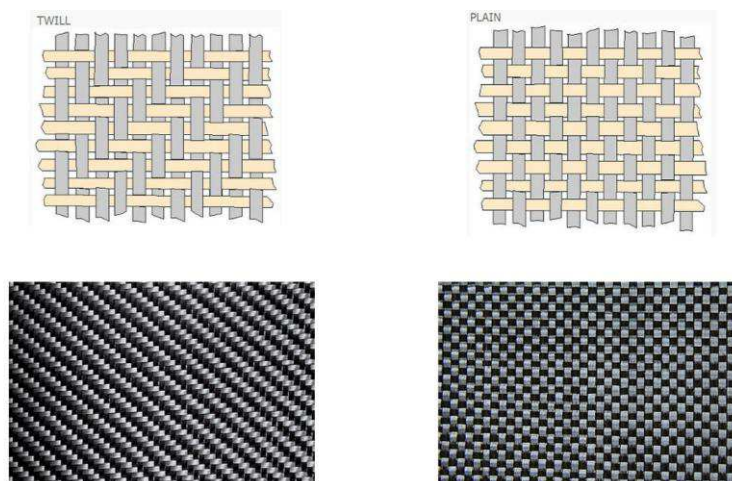
Obr. 14 – Karbonová kolejnice.

- **Technologie výroby vláken**

Uhlíková vlákna o rozměrech 5-8  $\mu\text{m}$  jsou navíjena ve svazcích po tisících, typicky 3K<sup>5</sup>. Svazky jsou spřádány do určitého vzoru, nejčastěji twill nebo plain. Vlákno použité karbonové trubičky je spřádáno formou plain. Patří mezi standardní jednoduché vzory, pevností je však dostačující. V otevřených koncích trubičky jsou vlepeny závitové vložky pro uchycení dorazu kolejiště (příčnicku). Obecně, veškeré modifikace karbonového profilu lze provádět pouze lepením. Broušením či vrtáním dojde k porušení vazby vláken a pevnost profilu se naruší. Upínací stativová ploška je lepena stejným způsobem (polymerním pojivem) [1-2].

---

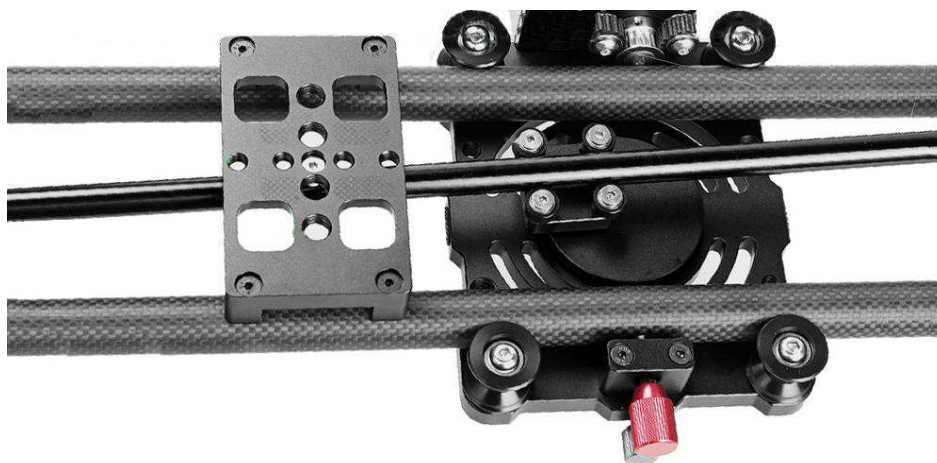
<sup>5</sup> K – určuje počet v tisících. Další vyskytující se velikosti 1K, 6K, 12K, 24K.



Obr. 15 – Vzory splétání svazků uhlíkových vláken (převzato z [2]).

### 2.2.2 Vozík

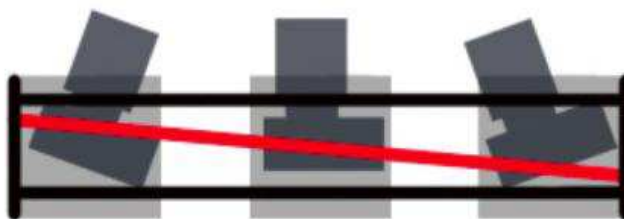
Obdélníková deska vozíku je v průřezu diagonál osazena axiálním válečkovým ložiskem z vrchní strany a ze spodní strany ložiskem kuličkovým. V ose ložisek je umístěn šroub pro upevnění fotoaparátu. Fotoaparát doléhá na vrchní miskou ložiska šroubově spojenou s miskou spodní.



Obr. 16 – Kolejnice, vodící tyč, zleva poté středová destička pro upnutí stativu, kamerový vozík a excentrické uchycení spodní misky ložiska pro rotační natáčení kamery (převzato z [3]).

Točivý moment na ložiska je vyvíjen excentrickým spojením spodní misky ložiska s tyčovým vedením umístěným pod kolejištěm v celé délce. Tyčové vedení lze vychylovat v celé šíři rozvoru kolejí tzv. jeho okrajové části v uložení příčnicku. Střed vodící tyče je uchycen v upínací destičce pro stativ. Nulové až diagonální nastavení úhlu (cca 20°) vodící tyče následně ovlivní způsob natáčení ložisek resp. fotoaparátu během přejezdu kolejiště.

Obecně, vychýlením získáme rotační pohyb sledující bod (v polovině dráhy přejezdu) před kolejištěm.



Obr. 17 – Vychýlení vodící tyče a vizualizace úhlového natáčení fotoaparátu (převzato z [3]).

Plynulý pohyb po kolejišti zajišťují rolny. Jedná se o kladku uzpůsobenou na vnější průměr trubičky-kolejnice. Profil obejmě z jedné poloviny a v sestavě čtyř kusů udrží vozík spolehlivě mezi kolejištěm. Valivý odpor je minimální a zakrytá ložiska v kladce jsou chráněna před nečistotami. Povrch kladky či kolejí lze jednoduše očistit a neulpívají zde nečistoty případně prach.

- **Příčnky**

Příčnky, nebo-li dorazy kolejiště, tvoří nosnou část konstrukce jsou vybaveny sklápěcími nožičkami s nastavitelnou výškou pro vyrovnání horizontu a protiskluzovými kulovými patkami. Karbonové kolejnice jsou z části zapuštěny v příčniku a garantují stejný rozvor kolejí v celé délce.

## 2.3 Změny konstrukce

Dosavadní koncept modelu Ashanks pracoval bez výjimky správně. Nicméně instalace motorizace pro možnost snímání časosběrů vyžadovala zásah do konstrukce.

Nový koncept zahrnoval integraci pohonu pro lineární posuv, pohon pro rotaci, řídicí elektroniku, krycí moduly elektroniky a uložení motoru. Mezikroky k dosažení cílů nebyly však vždy jednoduché. Často byly díly mezi sebou provázány a kroky bylo nutno pečlivě promýšlet, posléze realizovat.

### 2.3.1 Vodící tyč

V prvním kroku úpravy došlo k odstranění vychylovacích vodítek společně se středovým šroubem a následně vyjmutí vodící tyče. Šroubové spoje před demontáží bylo nutno nahřát horkovzdušnou pistolí pro povolení mezizávitového lepidla. Demontáží jsme získaly prostor pro osazení pohonu pod vozík, instalaci krabičkových modulů a dráhu tažného řemene.

### 2.3.2 Přestavba vozíku

Vodící systém tyče ve spodní misce ložiska neplnil již žádný účel. Došlo tedy k demontáži vedení i se spodní krycí miskou ložiska. Původní rozměrná ložisková miska byla, nahrazena pojistným šroubem, který nyní prochází celým závitem až k vrchní ložiskové misce. Šroub má jistý přesah přes hranu misky, tak aby mohl ukotvit foto příslušenství<sup>6</sup>.



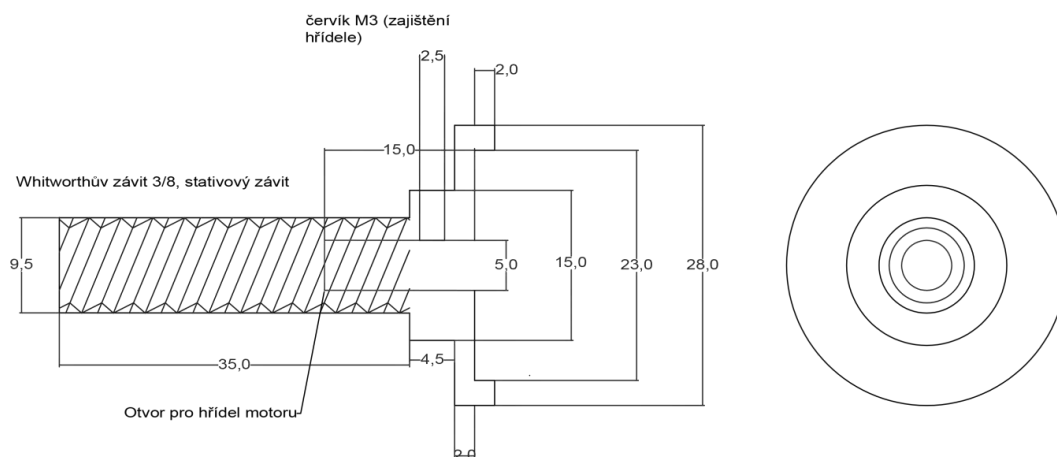
Obr. 18 – Půdorys vozíku s odejmutou horní miskou ložiska.

Výroba pojistného šroubu svírající misky ložisek byla neobvyklá typem závitu používající se u foto/video techniky. Závít vychází z tabulky whitworthových závitů. Načež se využívají průměry 1/4“ a 3/8“. Záleží na typu příslušenství i jeho hmotnosti.

---

<sup>6</sup> Zejména, stativovou kulovou hlavu, výklopnou destičku pro vertikální snímání či samotný fotoaparát.





Obr. 19 – Návrh soustruženého šroubu. Závitová část je umístěna v horní misce, zatímco spodní obvod šroubu je opřen o dolní ložisko. Dotažením vymezíme vůli misek. (uvedené hodnoty jsou v milimetrech).

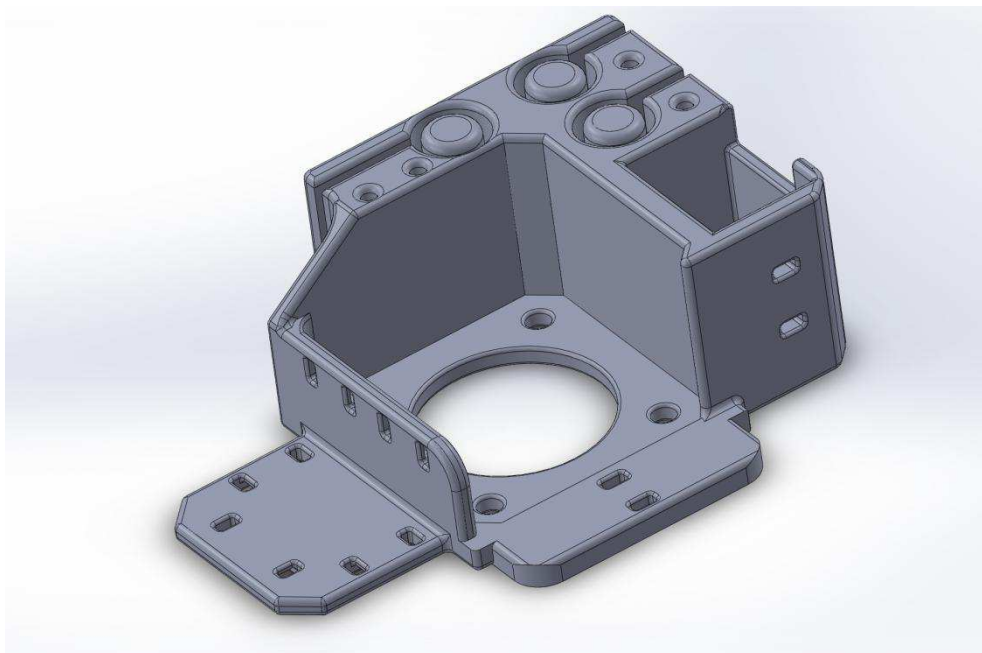
V této aplikaci<sup>7</sup> se použil šroub 3/8“ s dutou osou vysoustružen z hliníku. Do dutiny se vkládá hřídel motoru a zajišťuje se stavěcím šroubem (červíkem) proti prokluzu. Šroub ukrývá celou délku hřídele, přičemž tělo motoru lícuje se spodní stranou vozíku. Motor<sup>8</sup> je k vozíku chycen přes hliníkovou destičku. Destička kryje ložisko před nečistotami, jistí motor proti otáčení, drží modul přenášející pohyb řemenu na vozík spolu s výkonovou elektronikou pro spínání cívek motoru.

Vozíkový modul tvoří domeček motoru (motor rotace) doplněný spojkami pro uchycení tažného řemene a spínací elektronikou. Motor je objímán po celém obvodu do plné výše. Tím získává modul na kompaktnosti a pevnosti pro vertikální tažení břemene. Upnutí řemene je řešeno důmyslně přes rychlonavlékač spojky (speciální tvar v těle modelu), nevyžaduje žádné zajišťovací prvky. Zuby řemene se po obtočení válce vsunou (zakousnou) do sebe a v úzkém průchodu fungují jako zámek. Řemen ze zúženého prostoru nemůže vyklouznout. Vytvořená smyčka přenáší sílu na válec. Odejmutí řemene je jednoduché a není nutno nástrojů pro jeho sejmutí. Což je uplatnitelné pro manuální posuv vozíku bez mechanického brzdění motoru. V modulu nalezneme kapsu pro destičku elektroniky řídící spínání cívek motoru<sup>9</sup>. K ovládání je vyžadováno určitého počtu vodičů jenž lze snadno přichytit k povrchu pomocí stahovacích pásků.

<sup>7</sup> Šroub definovala horní miska ložiska.

<sup>8</sup> Motor vykonávající rotační pohyb těla videotechniky.

<sup>9</sup> Destička je zde umístěna kvůli délce vodičového spojení s motorem. Krátké vedení, nižší parazitní vlivy během přenosu, malé rušení.



Obr. 20 – Domeček motoru, rychlospojky pro řemen (v horním rohu lze vybrat ze dvou úrovní dle průměru řemenice motoru), hluboká kapsa pro DPS řízení motoru a připojení vodičů.

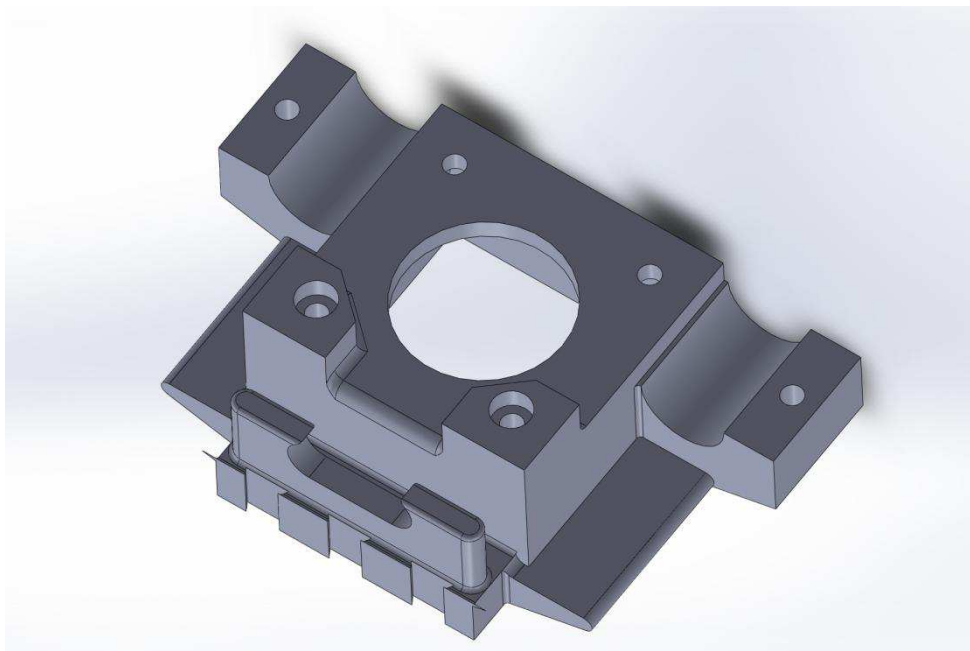
### 2.3.3 Pohon vozíku

Z popisu pohonu první generace výrobku viz první kapitola, umístění motoru hraje významnou roli pro těžiště celého výrobku. Je proto optimální motor umístit do osy konstrukce – kolejiště. Hmotnost motoru poté klade nižší nároky na konstrukci. Nedochozí zde ke krutu kolejnic. Uchycení motoru musí dále korespondovat s výškovým postavením hřídele k dráze tažného řemenu. Délka hřídele poskytuje jistou toleranci k doladění výšky řemenice. Kotvící domeček motoru (motor posuvu) je dimenzován na druhou nejhmotnější část konstrukce<sup>10</sup> a nejvíce namáhán bude při manipulaci zařízením.

Nesmíme opomenout ani na nutné řízení motoru, elektronika by měla být vždy v blízkém dosahu, tedy na stejném konci kolejí jako motor. Nelze tedy elektroniku použít jako protizávaží na opačné straně kolejiště.

---

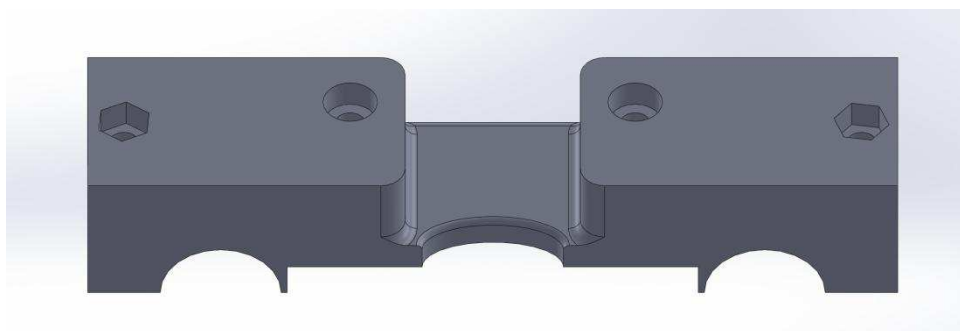
<sup>10</sup> První je vozík



Obr. 21 – Pohled zdola, v předu rybina, vsuvný zámek do příčnicku kolejiště s vykrojením pro kabeláž, dno motoru s otvorem pro přírubu s hřídelí a půlkulaté sáně doléhající na kolejiště.

Domeček motoru stejně jako v předchozím případě je modelován v programu Solidworks. Usazení je dimenzováno na typizovanou rozteč krokových motorů NEMA23. Lze volit z této škály nabízených druhů motorů a zvolit optimální poměr mezi váhou a krouticím momentem.

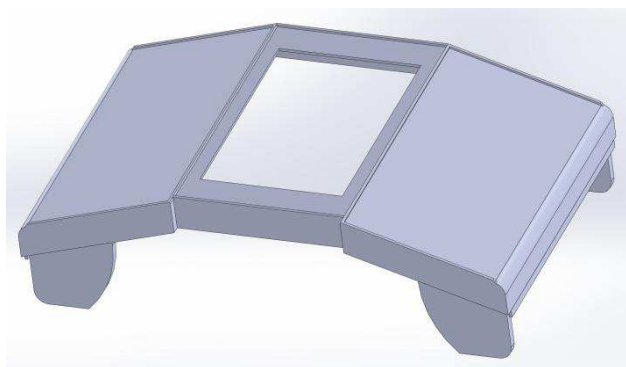
Modul domečku jedním koncem doléhá na kolejiště a druhým je zasunut do příčnicku kolejiště. Nehrozí tedy jeho posunutí vlivem tahu řemene či manipulace. Z důvodu nesené hmotnosti je modul posazen „uvnitř konstrukce“ na kolejišti. Zasahuje tedy délkou 72mm do dráhy kolejiště. Výsledná dráha se sníží na rozpětí 925mm a efektivní pojezdová dráha vozíku je 800mm (šíře vozíku je 125mm). V čele (u příčnicku) navazuje rybinový zámek pro krabičku nesoucí elektroniku. Ze spodní strany kolejiště je poté namontován díl objímající dohromady celou kolej a uzavírající stranu k vozíku.



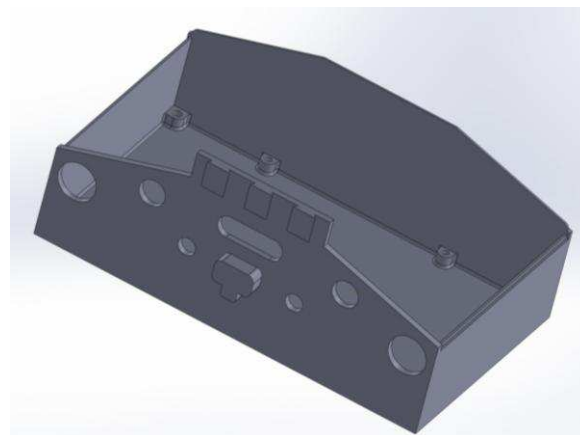
Obr. 22 – Spodní díl doléhající na koleje a zajišťující šroubově domeček motoru.

### 2.3.4 Krabička elektroniky

Díl byl původně modelován dohromady s domečkem motoru. Kvůli velikosti musel být však rozdělen. Vytvořením vazby pomocí klínových zubů resp. vybrání (rybina) vznikl mechanický spoj mezi díly a modely mohly být propojeny jako sestava dílů. Stěnu krabičky pokrývá soustava servisních otvorů s přístupy ke šroubům konstrukce slideru a průchodka pro kabeláž. Plocha dna krabičky byla koncipovaná, aby pokryla šíři hlavního plošného spoje a na výšku skryla spínací modul řídící motor posuvu. Víčko krabičky designově kopíruje tvar spojníku kolejiště. Víčko bylo děláno účelně symetrické z důvodu přístupu k zařízení z obou stran (dopad na ergonomii ovládacích prvků). Doléhající hrany krabičky i víčka mají vybrání pro zlepšení mech. soudržnosti a původně i zamýšlené vyšší ochraně před kapající vodou. Ovládací a vizuální prvky umístěné ve víčku, však plány narušily.



Obr. 23 – Krabička víčko, otvor pro display.



Obr. 24 – Krabička spodní díl, soustava servisních otvorů, průchod kabeláže, vystupující díl pasující do spojníku kolejí, klíny - mechanický zámek dílů.

### 2.3.5 Volnoběžná kladka

Přenáší moment motoru na vozík z protější strany kolejí. Zvolený průměr volnoběžné řemenice nemá žádný vliv na výsledný moment kladený na vozík. Záběrný moment je ovlivněn řemenicí na straně motoru.

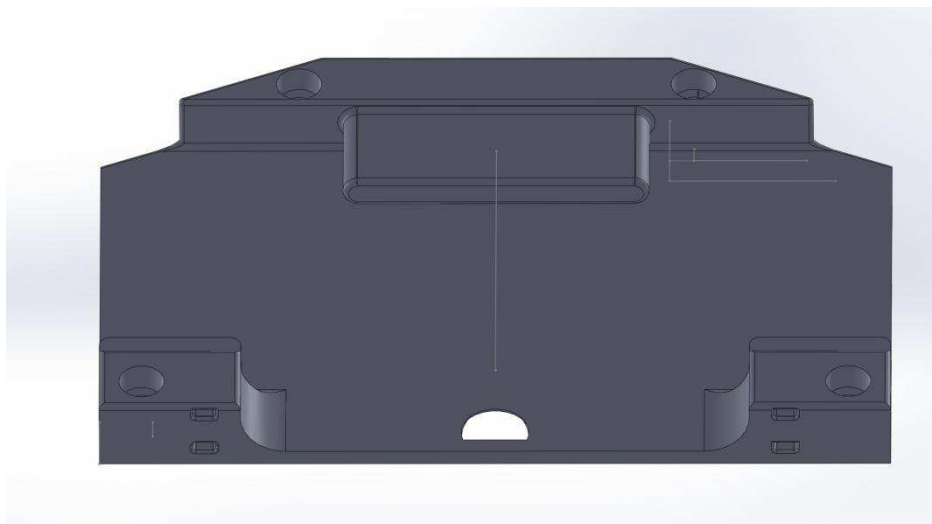
Kladkový nosič byl původně řešen pro jednu středovou řemenici většího průměru. Nicméně po vyzkoušení funkce byl řemen z části rozevírán motorem a docházelo k pnutí řemene. Pohyb vozíku byl omezen a těžce dojížděl ke kraji. Výměna řemenice za větší průměr byla vyloučena, zde by docházelo k přímé kolizi řemenice s motorem a opět omezení

dojezdu do kraje kolejiště. Vyvarováním se problémům s dojezdy je vyřešeno dvojicí řemeniček umístěných v krajích kladkového nosiče. Nyní řemen motor mívá v dostatečné vzdálenosti a vozík se bez obtíží přiblíží do úplného kraje kolejiště.

Nosič řemenic je vyroben z 2mm hliníkového plechu uzavřen do tvaru U s otevřenou roztečí 22mm (5mm rezerva je vyhrazena pro seřízení úrovně řemen – kladka). Řemeničky jsou nasazeny na 5mm ose otáčející se v ložiskách nalisovaných v plechu. Nosič je šroubovým spojením připevněn k spojníku kolejiště (využívá původních závitů po demontáži vodící tyče.)

### 2.3.6 Baterie

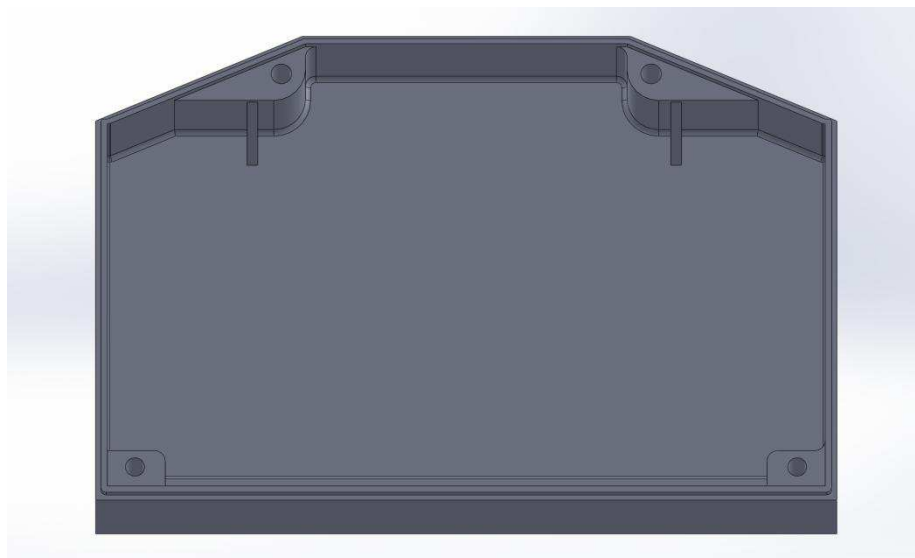
Koncept využití hmotnosti baterie bylo umístit články na protější stranu kolejí a vytvořit protizávaží případně kompenzovat alespoň z části hmotnost motoru. Vyvážení kolejí je důležité pro možnost upnutí zařízení na stativ. Budeme-li pracovat pouze s jedním stativem, což je žádoucí z důvodů mobility, hlava foto stativu vyrovná jen určité nesymetričnosti rozložení zatížení. Bohužel nežádoucí vlastností umístění baterie je vedení kabeláže přes celou délku kolejí k elektronice řízení. Průměr kolejnic je však dostatečný a rolny vozíku využívají jen vnější polovinu trubkovitého profilu. V tomto případě řešitelný nedostatek.



Obr. 25 – Víčko krabičky. Od shora, otvory pro šroubové spojení, vsuvný zámek do příčnicku kolejí (vysunutá stěna), průchodka pro kabeláž, otvory pro stahovací pásy vedení.

Vnitřní prostor krabičky je konstruován pro sadu 6-ti článků 18650. V případě potřeby je možné prostor upravit i pro 8 článků k dosažení větší kapacity, potažmo hmotnosti a lepšího vyvážení. Tělo a víčko krabičky k sobě doléhají obvodovým zámkem umístěným uprostřed

stěny. Zatěsněním šroubových a kabelových otvorů je krabička vodotěsná. Vrchní stěna kopíruje tvar příčnicku a je vybavena vsuvným zámkem totožně jako domeček motoru posuvu. Šířkou se krabička vejde přesně mezi osy nožiček (dva články na délku + šířka stěny krabičky). Po zasunutí krabičky baterie do zámků jsou všechny periferie na jedné konstrukci, získáme kompaktní zařízení a práce se zjednoduší – zefektivní.



Obr. 26 – Vnitřní prostory krabičky pro články 18650.

## 2.4 Zhodnocení konstrukce

Osazením veškerých komponentů zpět na konstrukci resp. montáží dílů nových zůstal původní koncept nenarušen. Výrobek byl jednoduše ovladatelný stejně jako výchozí konstrukce ASHANKS. Délka konstrukce mírně vzrostla o šíři krabičky elektroniky a bateriového modulu. Původně zamýšlená hladina dráhy řemene se posunula blíž k vozíku a byla tak upravena výška řemenic. Toleranční vymezení na hřídeli změnu výšky umožnilo.

### 2.4.1 Vodičové spojení

Holou mechanickou kostru zařízení bylo nutné protkat vodičovými komunikačními spoji i napájecími body zajišťující mimo jiné přenos informací mezi procesorem a řízení rotace hlavy vozíku s obsluhou spouště fotoaparátu. Postupně přibývajících kabelové svazky ve vnitřním obvodu kolejiště potřebujeme vhodně rozložit, aby nedocházelo k omezení pohybu vozíku v úzké části přejezdu (nad stativovou destičkou).

V prvním případě se jednalo o nepohyblivé statické kabelové vedení propojující koncový spínač a zdrojový přívod s hlavní elektronikou. Vedení bylo v celé délce lepeným spojem upevněno na kolej č. 1. Pouze v prostředku dráhy kolejí byla narušena izolace zdrojového kabelu pro napojení napájecího vedení k vozíku. Tato napájecí větev se pohybovala již volně dle pozice vozíku.

Kolej č. 2 nese komunikační spoje. Svazek dvanácti<sup>11</sup> vodičů je lepen na spodní a vnitřní část koleje nezasahující do dráhy rolny ani domečku motoru na vozíku. Soustava vodičů je opětovně vyvedena v prostředku a s daným průvěsem zavedena do vozíku napojena k silovému driveru motoru. Průvěs kabeláže se pohybuje mimo konstrukci. Původní experiment s vnitřním kabelovým vedením znemožňoval plynulý přejezd nad stativovou upínací destičkou.

### 3 Elektronika řízení – jednotky pohonu

Koncepce rozvržení většiny elektrotechnického vybavení bylo nezbytné plánovat již během stavby konstrukčních prvků a tvorbě 3D modelů. Hrubý výčet elektronických prvků se držel zásad, které vyplývaly z realizace první verze lineárního posuvu osvědčeného v provozu. Nicméně oproti minulé verzi, mělo vzniknout několik úprav a rozšířit působnost použití kamerového posuvu. Lineární pohyb vozíku bude doplněn o novou osu otáčení. Šroub ukotvení fotoaparátu je propojen s hřídelí motoru a umožňuje rotační pohyb. Během posuvu bude moci sledovat objekt (zachování objektu v záběru) nebo dělat panoramatické posuvy. Řízení motorů bude efektivnější s možností nastavení proudového omezení dle účelu posuvu horizontální / vertikální posuv.

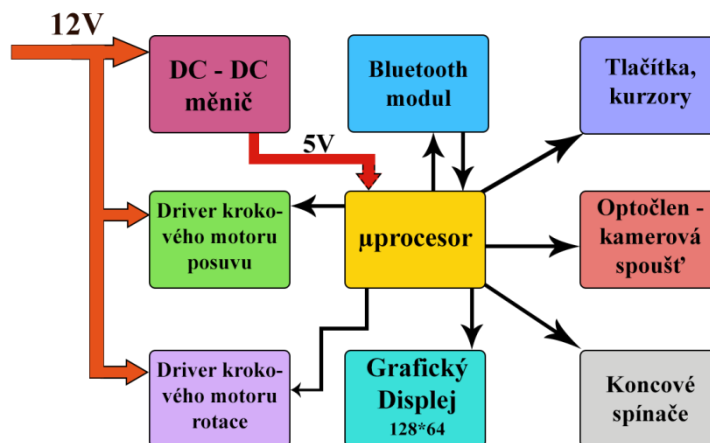
#### 3.1 Návrh řídicí elektroniky

K nastavení chodu motoru, obsluhy řídicích veličin a kontroly činnosti vyžadujeme vizuální prvky interpretující aktuální stav spolu s prvky spínacími poskytující vstupní informaci o změně. Návrh elektroniky zahrnoval grafickou jednotku displeje, přímá tlačítka nastavení, pohybové kurzory menu a výkonné pohonné jednotky manipulující s břemenem vozíku a rotací těla fotoaparátu. Tyto stěžejní body jsou doplněny následujícími pomocnými obvody pro činnost časosběrného posuvu fotoaparátu:

---

<sup>11</sup>Zpětná vazba natočení motoru, řízení motoru, spoušť fotoaparátu.

Koncovými spínači, detekujeme dojetí vozíku na konec kolejové dráhy. Optočleny, galvanicky oddělujeme spínání závěrky fotoaparátu z důvodů odlišné logické úrovně a vyšší bezpečnosti kamerové techniky. Bluetooth modulem, k bezdrátovému řízení vozíku, zvyšující rozsah možností obsluhy, kontroly scény a kompozice kamery.



Obr. 27 – Blokové schéma řídicí elektroniky.

### 3.1.1 Realizace

Realizace hlavní řídicí elektroniky se odvíjela od vybraného typu  $\mu$ procesoru související s počtem bran, funkcí pinů i možnostmi zapojení. Jádro kontrolující veškeré periferie patří do rodiny ATMEGA, typ 128A. Nízko napěťová varianta zde pracuje na standardní TTL napětí (5V), přesto v komunikaci s některým z modulů (bluetooth) je využíváno 3,3 V. Převládající většina bran je zakončena pinovým konektorem pro obsluhu vnějších periférií.

Motory jsou řízeny skrze modul, řešící správné spínání cívek motoru v cyklicky se opakujícím sousledu. K modulu jsou zpravidla vedeny tři logické vodiče dostačující k řízení motoru s plným krokem. Určení přibližné polohy vozíku, nebo-li nasnímání pohybu vozíku řešíme zpětnou vazbou. Generované pulzy od motoru jsou vedeny do komparátoru a posléze ke vstupům  $\mu$ procesoru k porovnání.

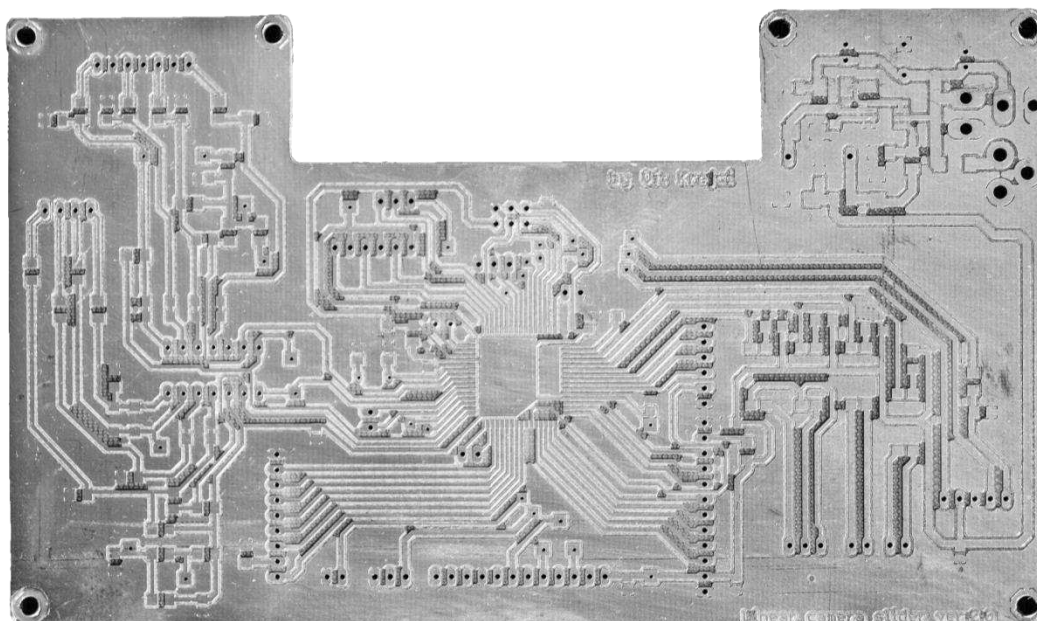
### 3.1.2 Výroba plošného spoje

Realizační a návrhové práce byly provedeny v programu Eagle. Propojení vývodů součástek probíhá ve dvou vrstvách (Top, Bottom). K frézovému opracování byl použit standardní oboustranný plošný spoj s 35 $\mu$ m folií. Izolační vzdálenosti, které je fréza schopna spolehlivě dodržet jsou  $\geq 0,3$ mm. Tvar desky plošného spoje není pravidelný, střed horní



hrany je ubrán z důvodů vsunutí modulu krokového motoru. V každém rohu je připraven montážní otvor k fixaci DPS v krabici. Potenciál GND v obou vrstvách vyplňuje nevyužitou plochu desky. Proudové zatížení 5V napájecí větve dle kalkulace odběru periférií by nemělo překonat hranici 45mA.

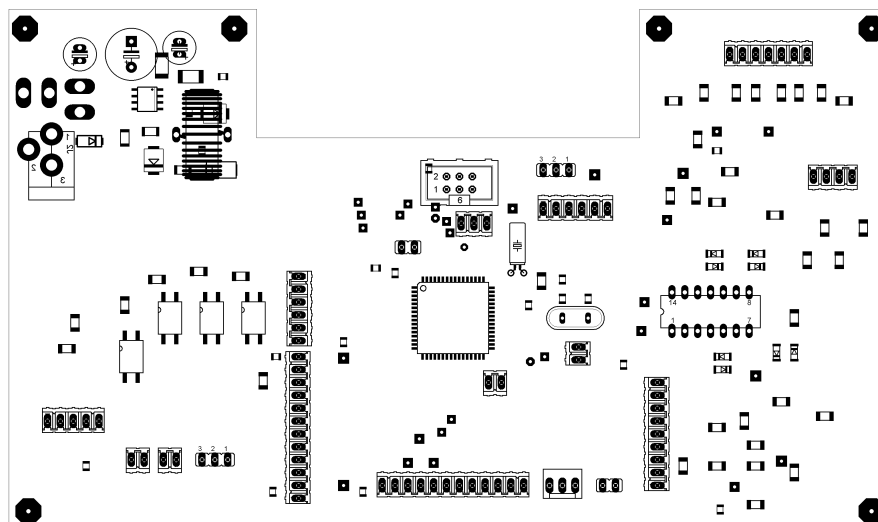
Ruční osazení součástek znamenalo postupné pájení jednotlivých modulů, aby se ověřila funkčnost a bezchybnost zhotoveného zapojení. Individuální obvod se měřením zkontroloval. Napájecí část nebo-li obvod měniče byl zapájen přednostně. Výstupní napětí bylo v úrovni 4,6V. Dělicí poměr napětí nebyl přesný, některak však nebránil v provozuschopném stavu. Dostupné napájecí napětí logických částí vedlo k ověření správného přiletování procesoru a zkontrolování komunikace s programátorem. Ostatní části byly zrevidovány pouze opticky a testovány jako celý DPS.



Obr. 28 – Plošný spoj, bottom vrstva.

### 3.2 Typy modulů

Z velké části je většina signálových cest procesoru vyvedena na konektorové piny připojující periferní moduly umístěny mimo plošný spoj. Majoritně jsou moduly propojeny interně v krabici. Několik svazků vodičů je však taženo dlouhým vedením až k vozíku (délka přes 1m). Dlouhé vedení připojujeme k plošnému spoji přes šroubové svorky.



Obr. 29 – Rozmístění součástek DPS.

V horním levém rohu DPS se nachází blok spínaného DC – DC měniče s kondenzátorovými filtry pro napájení spínaného modulu motoru. Pod blokem napájení jsou optočleny kontrolující fotoaparát a driver motoru. Střed DPS je zaplněn výstupy k různým perifériím spolu s procesorem. Pravá část je vyčleněna komparátoru k vyhodnocení souslednosti pulzů z motorů.

### 3.2.1 Napájecí měnič

Využívá rozšířený integrovaný obvod MC34063 zapojený dle katalogového listu v režimu step-down. Spínaný stabilizátor snižuje vstupní napětí baterie 12V na pracovní hodnotu řídicí logiky 5V. Proudové nastavení zdroje je omezeno na 70mA paralelní kombinací rezistorů R7 a R14. Minimální velikost indukčnosti dle výpočtu odpovídá 81  $\mu\text{H}$ . Zvolena dostupná hodnota je 100 $\mu\text{H}$ . Velikost výstupního napětí odpovídá zvolenému poměru rezistorů (R15, R13) v napěťovém děliči. Vybíráme-li z normalizované řady hodnot, nejbližším zvoleným kompromisem rezistorů došlo k poklesu napětí o čtyři desetiny voltu. Měřením na prázdko je svorkové napětí 4,6V, zatímco při testované zátěži 72mA pokleslo napětí k 4,25V.

Příklad výpočtu indukčnosti cívky<sup>12</sup>[15]:

$$L_{(min)} = \left( \frac{V_{in(min)} - V_{sat} - V_{out}}{I_{pk(switch)}} \right) * t_{on(max)} \quad (1)$$

<sup>12</sup> Pro získání všech parametrů bylo využito předdefinovaného tabulkového kalkulátoru.



$$i_{vin} = \frac{U_{mot}}{R_{vin}} * (1 - e^{\lambda t}) \quad (2)$$

$$u_L = U_{mot} * e^{\lambda t}$$

$$\lambda = -\frac{R}{L} = \frac{-1}{\tau} \quad (3)$$

Oba následující obvody mají totožnou funkci a obdobný způsob řízení. K ovládní posuvu motoru stačí dva logické vodiče. DIR k výběru směru rotace rozhoduje logická úroveň 1 nebo 0. STP volí rychlost posuvu (počet kroků) reprezentovanou změnu „krok“ vyvolá vzestupná hrana vstupního signálu. Další ovladatelné vstupy mohou sloužit pro správu kroku motoru (mikrokrokování), režimy úspory, vyřazení obvodu, atd. Aktivací pinu ENB se deaktivuje H-můstka, výstup se přepne do vysoké impedance a vstup STEP je ignorován. [18]

- **Řízení posuvu – TB6560**

Modul s integrovaným obvodem Toshiba TB6560 je řídicí obvod pro bipolární krokové motory s proudovým rozsahem až 3A na fázi vinutí. Provozní rozpětí napětí je 10 – 34V s doporučenou hodnotou 24V k dosažení optimálních parametrů. Rezistence H-můstku jedné fáze při sepnutém stavu a odběru 1,5A je v toleranci 0,35 – 0,5Ω. Generované ztráty během spínání jednotlivých fází a sepnutém-vodivém stavu, jsou odváděny pasivním chladičem dodávaným s modulem. Plošný spoj modulu je osazen několika přepínači k nastavení proudových parametrů (dle nominální proudové hodnoty fáze vinutí), přepínáním mezi měřicími rezistory. Spínačovým blokem definujeme rovněž procentní hodnotu klidového proudu (jestliže motor stojí), řízení velikosti kroku - mikrokrokování a rychlost odproudění cívky vinutí (rychlé otáčky hřídele, nepříjemné zvukové projevy na vysokých frekvencích) [19].

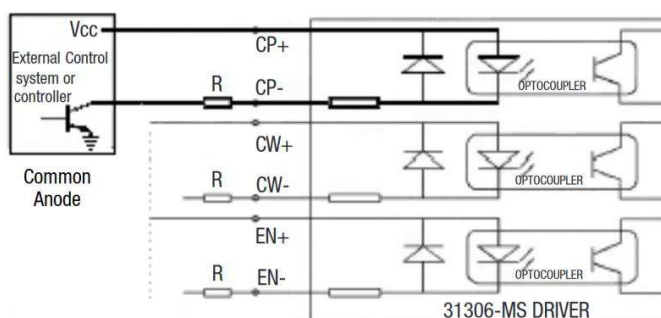
Tab. 6 – Proud vinutím fáze motoru (running current) [20].

(A)	0.3	0.5	0.8	1	1.1	1.2	1.4	1.5	1.6	1.9	2	2.2	2.6	3
SW1	OFF	OFF	OFF	OFF	OFF	ON	OFF	ON	ON	ON	ON	ON	ON	ON
SW2	OFF	OFF	ON	ON	ON	OFF	ON	OFF	OFF	ON	OFF	ON	ON	ON
SW3	ON	ON	OFF	OFF	ON	OFF	ON	ON	OFF	OFF	ON	ON	OFF	ON
S1	ON	OFF	ON	OFF	ON	ON	OFF	ON	OFF	ON	OFF	ON	OFF	OFF

Tab. 7 – Klidový proud, mikro-krokování motoru, rychlost odproudění fáze vinutí.

Stop Current	Excitation Mode	Decay Setting
<b>S2</b> 20% ON 50% OFF	<b>Step S3 S4</b> whole OFF OFF half ON OFF 1/8 ON ON 1/16 OFF ON	<b>S5 S6</b> 0% OFF OFF 25% ON OFF 50% OFF ON 100% ON ON

Vstupní řídicí logické úrovně STP, DIR, ENB jsou optočleny galvanicky odděleny od vyššího napětí. Světelný vysílač optočlenu je zapojen se společnou anodou ke zdroji napětí a katoda je jednotlivě spínána mikrokontrolérem.



Obr. 31 – Vstup optronu, zapojení se společnou anodou (převzato z [21]).

Modul je umístěn rovnoběžně se stěnou krabičky doléhající na dno v rezervovaném prostoru odebrané plochy DPS hlavní řídicí elektroniky. Pozice modulu byla vybrána z důvodu možnosti osazení chladiče mimo prostor krabičky. Venkovní proudění vzduchu zajistí lepší ventilaci a chlazení. V krabičce získáme taktéž více prostoru pro ostatní periférie a kabeláž. Originální délka (30 cm) přívodních vodičů motoru dosáhne ke svorkám modulu s rezervou a splní předepsané zapojení.

- **Řízení rotace – DRV8825**

Rozměrově velice malý modul z poloviny obsazen plochou integrovaného obvodu DRV8825 vychází z oblíbených řad driverů (A4988) pro Arduino. Proudové možnosti buzení dosahují k hodnotě 1,5A na jednu fázi bez přídavného chlazení čipu. S chlazením lze proudové zatížení zvýšit až ke 2,2 A. Nastavení proudového omezení vinutí motoru je realizováno trimrem na DPS. K měření odebíraného proudu nestačí sledovat pouze proudový odběr na zdroji, ale musíme sledovat proud fází. Efektivní hodnota napětí cívky není shodná



způsobem i přes bluetooth adaptér. Jedno zařízení pracuje v konfiguraci master, řídicí tok dat a druhá jednotka v slave režimu. Přenos informací mezi zařízeními začne probíhat po úspěšném párování bluetooth adaptérů. Výchozí přenosová rychlost modul je stanovena na 9600Bd. Skutečná rychlost dat je vždy menší, v řetězci odesílaných dat 8 bitů figurují rovněž řídicí startbit, stopbit popř. paritní bit. Z rozměrových dispozic zařízení (12,7mmx27mm), odrážející se na velikosti antény, se dosah signálu pohybuje v okruhu 10m. Spotřeba zařízení napájeného 5V je při navázaném spojení 8mA. Během párování odebíraný proud roste k 35mA.[23-24]



Obr. 33 – Bluetooth sériový adaptér HC-05.

### 3.3 Pohon

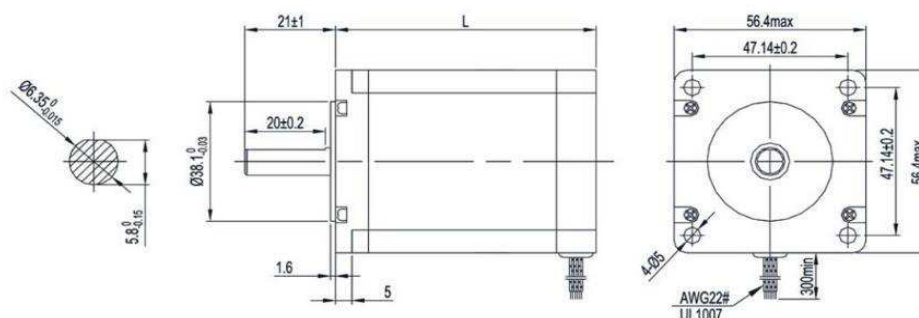
Rozhodujícími faktory k výběru optimální pohonné jednotky je několik. Příznaky jsou totožné s předchozí verzí posuvu a některé zde zopakujeme. Z principu chodu motoru vycházející z požadavku na práci časově náročného snímání je tažené břemeno neustále zastavováno a uváděno do pohybu v malých přejezdových úsecích. Výsledek tohoto nekontinuálního pohybu břemene jsou vysoké nároky na rozběhový moment motoru. Krokový motor tyto potřeby rozjezdů a brzdění uspokojivě zvládá, s opakující přesností založenou na digitálním řízení. K tažení nákladu je hřídel připojena přímo bez převodování. Pohon objemově i hmotnostně dále nenarůstá, je snadno vyměnitelný za podobný typ s jinými parametry a natočení tažné hřídele odpovídá natočení rotoru v motoru (odvození polohy břemene).

#### 3.3.1 Manipulátor vozíku

Pohon vozíku byl vybírán s ohledem na hmotnost, rozměry, finanční náklady a především točivý moment. Vyvinutý moment motoru je koncipován i pro vertikální zdvih tělesa, zde

jsou nároky mnohem vyšší než u posuvu horizontálního, kde váha vozíku je nesena kolejiemi a motor překonává pouze tření koleček vozíku.

Krokové motory, dnes výhradně používané bipolární, obecně dělíme dle rozměrů příruby (povrch těla motoru s vyvedenou hřídelí) označovanou NEMAxx (NEMA17, NEMA23, atd.). S velikostí příruby z pravidla roste točivý moment (průměr rotoru s magnetickými dipóly se zvětšuje). Dle typu vinutí cívek fáze a délky motoru (definována je pouze příruba) se parametry mohou různě lišit.



Obr. 34 – Šasi krokového motoru NEMA23, rozměrové dispozice (převzato z [27]).

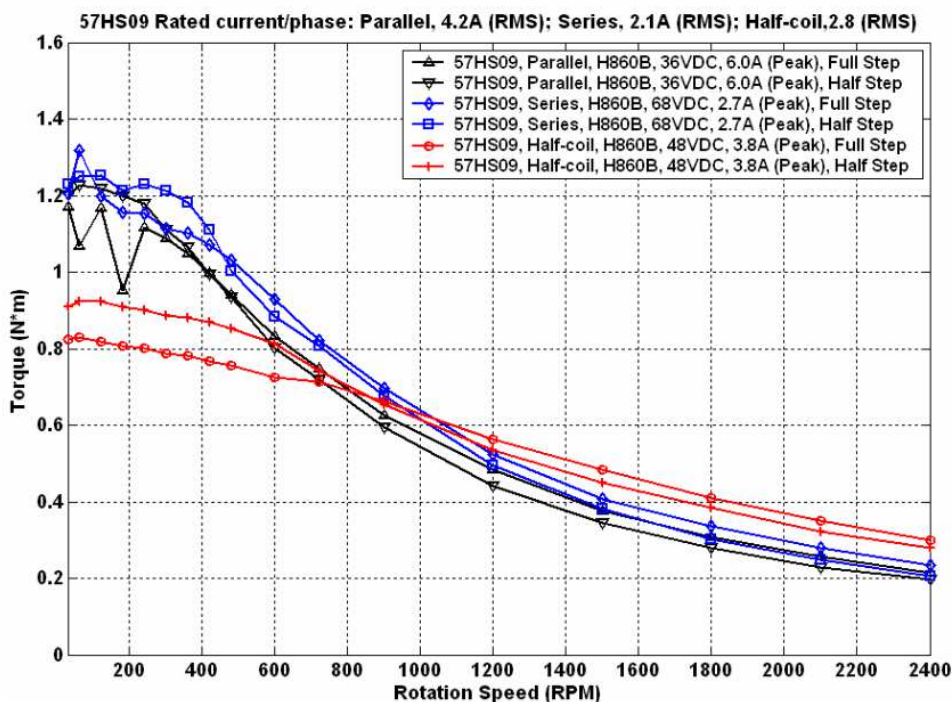
- **Výpočet momentu motoru**

K určení dostatečně výkonného motoru změříme použitou řemenici, zvážíme vozík se stativovou hlavou, fotoaparátem a objektivem. Spočtený moment viz tab. 8 je brán jako nejnáročnější - vertikální zdvih. Točivý moment krokových motorů je mimo jiné citlivý na provozované otáčky. Rostoucí počet ot/min negativně ovlivňuje výkon motoru. Vybíráme-li pohon, nemůžeme se spolehnout jen na tabulkové údaje, ale ideálně znát i plánované provozované rozpětí otáček a dle grafu závislosti krouticího momentu na otáčkách zvolit příslušný motor.

Tab. 8 – Moment motoru, výpočet.

Položka	Změřené hodnoty	Upravené hodnoty	Moment motoru
Váha vozíku s foto vybavením a fotoaparátem	2,85kg	$F = m \cdot g$ $F = 2,85 \cdot 9,82$ $F = 0,5339 \text{ Nm}$	$M = r \cdot F$ $M = 0,54 \text{ Nm}$
Poloměr tažné řemenice	19mm	0,019m	





Obr. 35 – Závislost krouticího momentu na otáčkách. Jedná se o graf motoru parametrově shodného s 23HS5628 (převzato z [25]).

$$\varphi_{kroku} = 1,8^{\circ} \quad (4)$$

$$\frac{360^{\circ}}{1,8^{\circ}} = 200 \text{ pozic}$$

$$\begin{aligned} f &\cong \omega \\ 200\text{Hz} &\cong 1\text{ot/s} \\ 2600\text{Hz} &\cong 780\text{ot/min} \end{aligned} \quad (5)^{14}$$

Zvolený dvojfázový krokový motor NEMA23 23HS5628 má vyvedeny 4 vodiče, každý pár pro jednu fázi. Cívky vinutí jsou spojeny interně. Tabulková velikost záběrného momentu je 1,26Nm, zdvih vozíku je dimenzován dostatečně s rezervou a je schopen vést těžší techniku nebo pracovat s rychlými přejedy. Dvojnásobné naddimenzování krouticího momentu je vykoupeno hmotností 0,7kg těla motoru. Preference je kladena k vertikálnímu zdvihu.

Tab. 9 – Parametry krokového motoru posuvu vozíku (převzato z [26]).

Model	Step Angle (deg)	Motor Length (mm)	Rated Voltage (V)	Rated Current (A)	Phase Resistance (Omh)	Phase Inductance (mH)	Holding Torque (Nm)	Motor Weight (kg)
23HS5628	1,8	56	2,5	2,8	0,9	2,5	1,26	0,7

<sup>14</sup> 780ot/min je horní hranice technických možností driveru ve vertikálním posuvu z hlediska jeho napájení.

Tab. 10 – Pořadí zapojení vodičů fází.

Modrá	Žlutá	Zelená	Červená
A+	A-	B+	B-

### 3.3.2 Manipulátor těla fotoaparátu

Koncepce pohonu rotace osy fotoaparátu je navrhována pouze pro horizontální posuv. Experimentálně i snímání v náklonu pod úhlem  $45^\circ$ . Zatížení kladené na motor se může zdát malé, ale jelikož je motor umístěn přímo v ose fotoaparátu, páka na motor je nezanedbatelná. Motor byl vybírán přednostně dle rozměrových dispozic, aby prošel nad spojovací středovou upínací destičkou a nenavýšoval výrazně hmotnost vozíku.

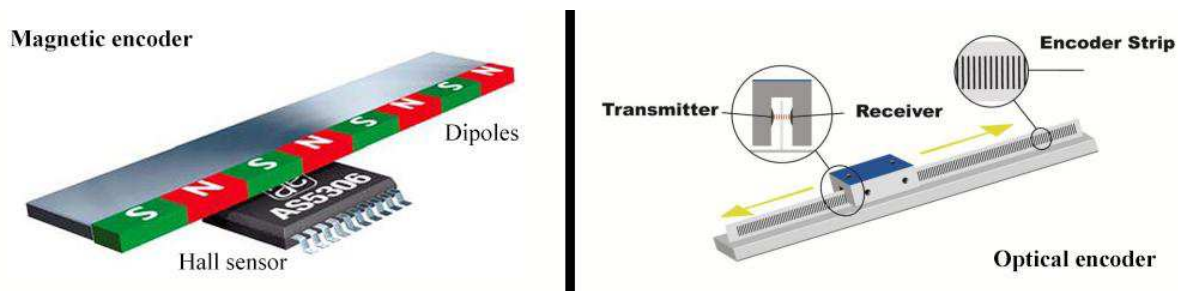
Vybrán byl krokový motor NEMA17 17HS2408s délkou těla 28mm, hmotností 150g a proudovou zatížitelností 0,6A. Hřídel motoru byla zkrácena z důvodu nasunutí vysoustruženého pojistného šroubu pro lícování příruby s domečkem. Jemné docentrování šasi motoru s osovým pojistným šroubem snížilo tření mezi ložinami vozíku a motoru.

Tab. 11 – Parametry krokového motoru rotace fotoaparátu (převzato z [27]).

Model	Step Angle	Motor Length	Rated Current	Phase Resistance	Phase Inductance	Holding Torque	Detent Torque	Motor Weight
	(deg)	(mm)	(A)	(Omh)	(mH)	(Nm)	(Nm)	(kg)
17HS2408	1,8	28	0,6	8	10	0,12	0,016	0,15

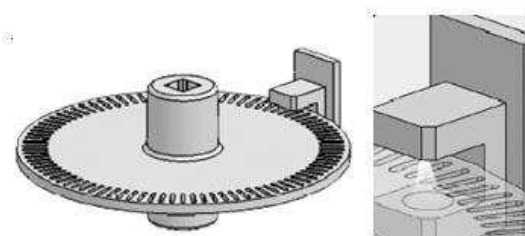
### 3.3.3 Zpětná vazba

K zjišťování polohy vozíku a smyslu natočení osy fotoaparátu slouží zpětná vazba. Původní plán určování pozice vozíku spočíval v použití magnetického enkodéru, pásku se střídavě orientovanými dipóly. Změnu orientace by detekoval Hallův snímač a inkrementoval počet změn k přejetí dráhy. Pásek by se nalepil na vnitřní stranu koleje, nebo na vnější stranu řemene se snímačem na vozíku. Nicméně dostupné magnetické pásy byly bohužel určeny pro odečítání extrémně přesné polohy v průmyslových strojích v jednotkách  $\mu\text{m}$ . Od toho se taktéž odvíjela vysoká cena převyšující cenu konstrukce posuvu. Magnetický pásek mohl být nahrazen ekvivalentem pásku čárového k optickému čtení polohy, avšak pásek by bylo nutné umístit do prostoru kvůli nezbytnosti čtecí hlavy objímající pás (světelný zdroj, fotocitlivý snímač). Lze narazit i na pásy reflexní se snímačem i vysílačem světelného paprsku na jedné straně pásu, opět se však pohybujeme v systémech profesionálního použití.



Obr. 36 – Magnetický a optický lineární snímač polohy vychází ze snímání změn detekující na pásku. Rozlišení pásku odpovídá rozlišení celého systému. Magnetické čtení je z hlediska údržby spolehlivější (převzato z [29 - 30]).

Kotoučové enkodéry nelze využít v případě motoru rotace osy vozíku (celé tělo je integrováno v domečku a nepřístupno z venčí). Motor posuvu má speciální podmínku na průměr enkodéru. Čárový kód by začínal až za úrovní tažné řemenice, ideálně od 44mm.



Obr. 37 – Optický enkodérový disk (převzato z [31]).

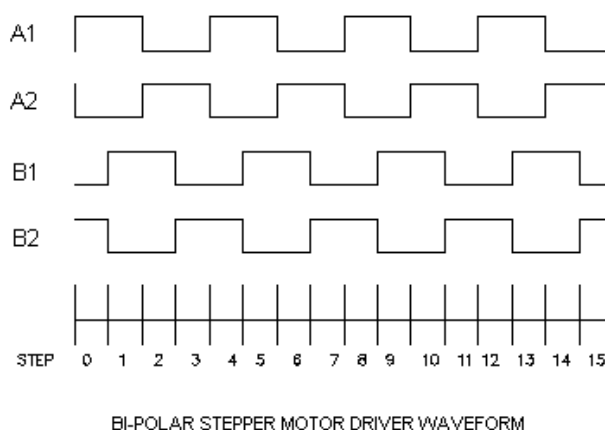
- **Motorický posuv vozíku**

Přesnost pohybu vozíku v kolejišti není kritická a lze se spolehnout na relativní polohu danou počtem pulzů odeslaných z MCU do driveru motoru. Vozík je jen nutné dostat do výchozí pozice, odkud bude znám počet kroků ke kraji kolejí (k příčnickům). Pomalé rychlosti posuvu snižují riziko kolize s okolním předmětem, prokluzu rotoru a ztrátu magnetické vazby mezi zuby rotoru - statoru. Jízda bez asistence zpětné vazby je kontrolována koncovými spínači zastavující vozík v krajních polohách a zároveň obnovující informaci o počtu kroků k přejetí k opačnému kraji (defaultní pozice).

- **Čtení polohy odvozené od pulzace krokových motorů.**

Významnějším řešením je zavedení tzv. „trekování“ zaznamenávání polohy os motorů během manuálního posuvu vozíku. Otáčením hřídele při rozpojených kontaktech motoru, v tomto případě přepnutí H-můstku do stavu vysoké impedance naměříme na svorkách generované napěťové špičky. Jednotlivé pulsy reprezentují mechanické otočení hřídele o  $1,8^\circ$ .

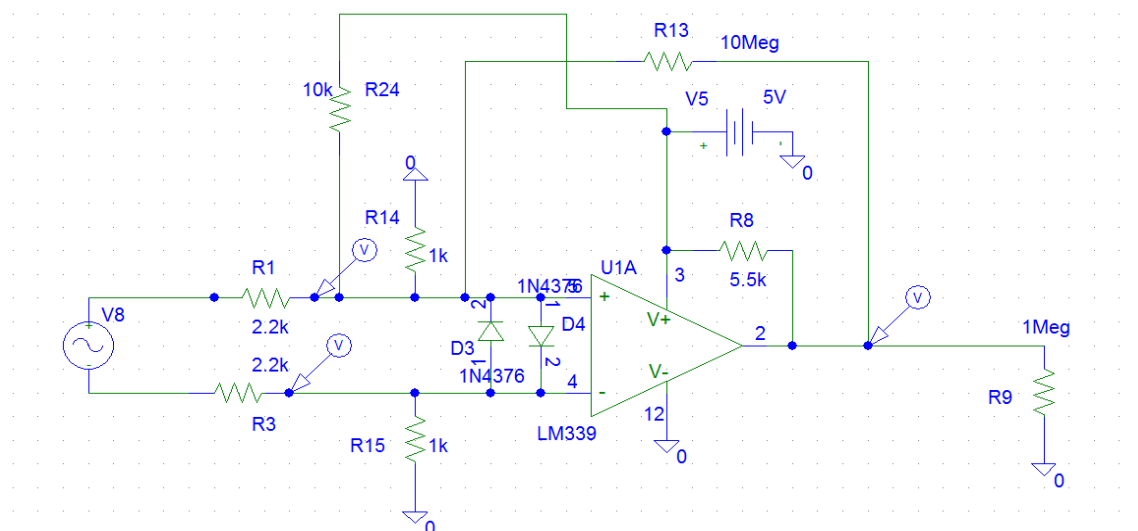
Tímto inkrementováním jednotlivých pulzů jsme schopni zopakovat pohyb driverem a rovněž určit dle překrývání pulzů směr otáčení. Princip načítání impulzů bude využit pro zachycení ideální uživatelsky zvolené přejezdové kompozice. Vybrání vhodné kompozice je spojeno s ručním posouváním vozíku, experimentování s ohniskovou vzdáleností objektivu, nastavením expozičních hodnot, atd. Hledanou přejezdovou kompozici zopakujeme v režimu načítání pulzů a s dojezdem ukončíme. Se získanou početní informací jsme schopni pohyb motoricky zopakovat v rámci mnohem delšího časového intervalu a vytvořit časoběrný videozáběr.



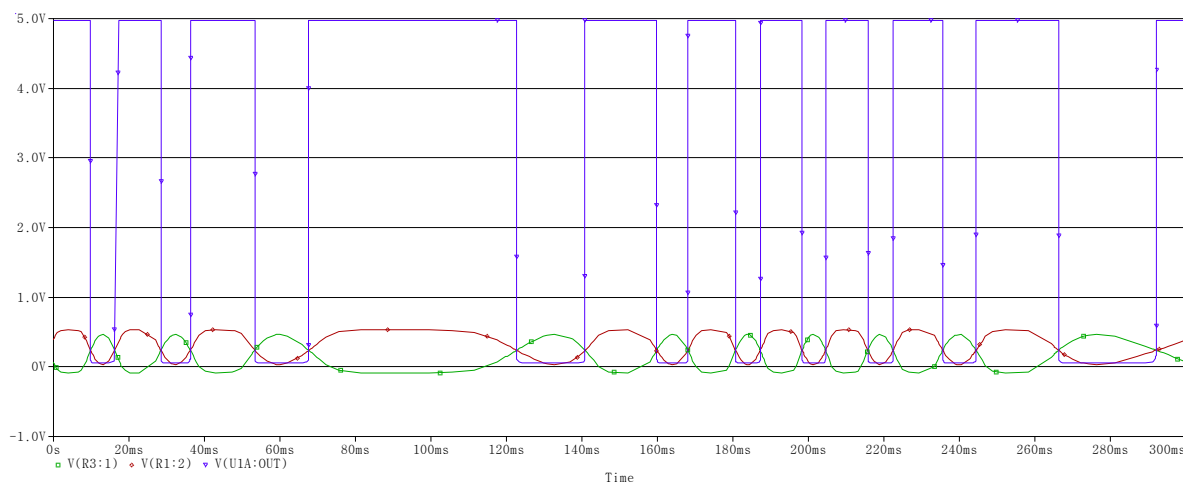
Obr. 38 – Pulzování driveru k roztočení hřídele. V podobném smyslu je generovaný pilový signál nezátíženého motoru při mechanickém otáčení hřídele (převzato z [33]).

- **Načítání impulzů**

Generovaný signál každé fáze motoru je veden na vstup komparátoru. Vstupní rezistence vylučuje proudový tok z H-můstek skrze pomocné obvody. Napěťový dělič upravuje vstupní úroveň napětí na 0,5V. Vstup komparátoru je chráněn rychlými diodami, pokud by došlo k zanesení vysoké napěťové špičky. Zpětná vazba výstupu je zpražena 10M odporem. Hodnota napětí kladného vstupu je uměle zvýšena 10k rezistorem pro rozšíření komparovaného pulzu. Pomocný pull-up rezistor vytváří na výstupu logickou úroveň přivedenou k MCU.



Obr. 39 – Schéma zapojení komparátoru, v programu LTspice Schematics.



Obr. 40 – Simulace obvodu komparátoru LM339. S nastavenými vstupními hodnotami naměřeny osciloskopem na výstupu motoru.

Jeden motor obsadí čtyři vstupy (2 komparátory) a získané dva signály porovnává mikroprocesor. K získání vzdálenosti jsou pulzy pouze inkrementovány a k určení směru otáčení je vzestupná hrana signálu\_A srovnávána s druhým signálem\_B. Signály jsou vzájemně otočeny o 180°. Pokud signál\_B je protínán v logické 1 vzestupnou hranou\_A. Točí se motor jedním směrem. Pokud v log. 0, směr je opačný. Naměřené zpracované hodnoty odešleme do driveru motoru a provedeme činnost posuvu.

### 3.4 Zdroj napájení

Výstupní napájecí napětí adaptéru se může pohybovat v rozpětí 11 – 15V. V aplikaci se však budeme zabývat primárně přenosnými zdroji čili akumulátory. Sekundární alkalické

články s malou kapacitou a výstupními proudy nejsou vhodné, rovněž i olověné akumulátory pro vysokou hmotnost. Pro napájení driverů motorů byly vybrány dostupné LiIon články 18650, používány hlavně v přenosných počítačích, RC modelech a powerbankách. Jsou kvalitním zdrojem energie, výkonné a lehké. Mají nízké samovybíjecí proudy a pokryjí nízké až střední odběry energie. Nominální napětí článku je 3,6V, maximální napětí plně nabitého článku pak 4,1V. LiIon akumulátory jsou z pravidla vybaveny kontrolní elektronikou sledující několik parametrů. Články jsou citlivé na nízkou hladinu napětí „podvybití“ a hodnoty pod 3V jsou vyhodnoceny k odpojení. Rovněž během nabíjecího cyklu je napájecí úroveň nutno stabilizovat na 4,1V s přesností  $\pm 0,05V$ . [28]

Baterie je složena ze sérioparalelní kombinace 6-ti článků, tzv. 3S (paralelní dvojice v tříčlenné sérii) s výslednou kapacitou 5200mAh. U každé paralelní dvojice je měřen potenciál ať už pro ochranu logiku elektroniky či optimální nabíjení battery packu pomocí LiIon dobíjecí stanice s balancérem. Balancér zajišťuje správnou úroveň napětí pro každý článek, je-li hodnota vyšší, dojde k připojení odporové zátěže a poklesu napětí. Chytré řízení nabíjení má pozitivní vliv na životnost článků baterie a zaručuje nabití článku na jeho plnou kapacitu. Proudový odběr z baterie se v průměru pohybuje kolem 0,7A v závislosti na nastavení driverů motorů a hmotnosti břemene.



Obr. 41 – Odkrytí pouzdra baterie, LiIon články 18650 s ochranou elektronikou.

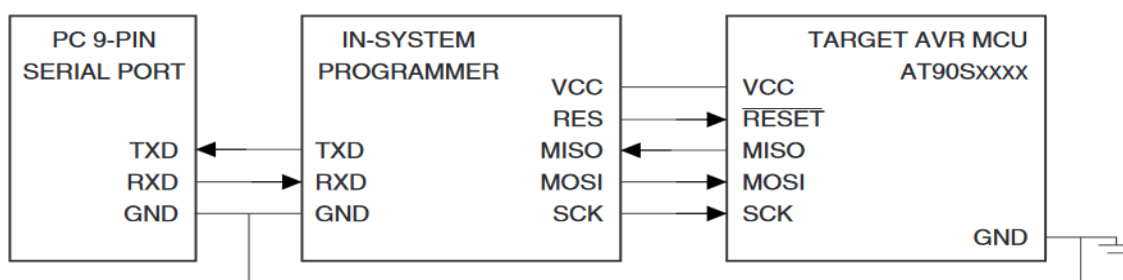
## 4 Programování

Psaní zdrojového kódu a nastavení veškerých funkcí  $\mu$ procesoru probíhá ve vývojovém prostředí CodeVisionAVR spolupracující s mikrokontroléry Atmel. Modelová řada mikroprocesorů ATmega128A disponuje 64 vývody uzavřena v pouzdře typu TQFP. Výstupy

a speciální funkce pokrývá 7 bran, z nichž brána G je poloviční. Veškeré vývody s výjimkou dvou vstupů jsou vyvedeny na desku.

## 4.1 Oživení

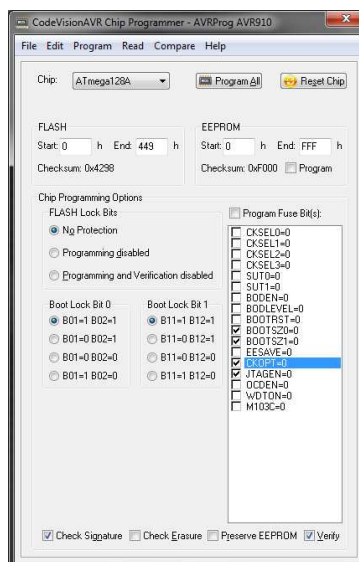
Komunikace s procesorem je navázána prostřednictvím programátoru PonyProg AVR 910. Data jsou přenášena sériovou linkou od PC k programátoru protokolem SPI (Serial Peripheral Interface) a programátor přeposílá data ISP (In-System Programming) protokolem k cílovému  $\mu$ procesoru. Jednotlivé bity jsou odeslány se synchronizačním pulzem SCK.



Obr. 42 – Způsob komunikace PC, programátor, cílový procesor (převzato z [33]).

### 4.1.1 Fuse bity

Navázání správné funkce komunikace ověříme již v programovém rozhraní CodeVisionAVR příkazem na rozpoznání cílového mikrokontroléru. Potvrzením správného modelu MCU se přesuneme k nastavení fusebitů. Během nastavování „pojistkových bitů“ věnujeme pozornost změně značení, log. 0 je prezentována jako „programováno“ a naopak. Bity CKSEL 0-3 Přepneme z defaultně nastaveného interního RC oscilátoru na vnější krystalový s taktem 16Mhz. V datasheetu nalezneme příslušnou kombinaci bitů CKSEL měnící nastavení kmitočtu a zdroj kmitů. S výběrem horní taktovací frekvence rovněž přepneme CKOPT bit měnící mód zesílení kmitů oscilátoru na rail-to-rail, signál je více imunní vůči rušení. Bity SUT 0-1 zůstanou v log.1 pro časovou prodlevu 64ms po restartu MCU.  $\mu$ procesor je z výroby nastaven do kompatibilního režimu rodinného modelu ATmega 103 a některé funkce nejsou v tomto módu dostupné. Proto tento režim fuse bitem M103C vypneme. Správně vyplněnou tabulku nahrajeme do zařízení.



Obr. 43 – Programovací okno, nahrání softwaru do paměti flash MCU. Nastavení fuse bitů.

### 4.1.2 Nastavení bran

Založením nového projektu a zvolením patřičné procesorové řady se otevře okno s jednotlivými volbami nastavení. Mezi které řadíme i definování bran. Veškeré spojení se spínacími prvky je definováno jako vstupní (IN) se zapnutými pull-up rezistory. Tlačítka jsou napájena z MCU vysokou úrovní a z druhé strany uzemněny. Stisknutím tlačítka dojde k uzemnění, úroveň napětí skočí k nule, proud je omezen pull-up rezistorem (obecně 10k). Zaznamená-li MCU hodnotu blízké 0V, registruje stisk tlačítka.

Synchronizace spouště fotoaparátu a driver posuvu vozíku jsou galvanicky odděleny od obvodu optočleny. Optrony jsou zapojeny se společnou anodou, k sepnutí členu uzemníme vstup katody. Potřebné porty jsou nastaveny jako výstupní s výchozím stavem log. 1.

### 4.1.3 Nastavení hodin

Nastavení hodinového kmitočtu je umožněno v položce čítač/časovač viz obr. 44. V podokně Timer0, jenž pracuje jako 8-bit čítač, vybereme zdroj hodin z pinu TOSC1 na který je připojen „hodinový“ krystal 32 768Hz. Dělič kmitočtu nastavíme na hodnotu 128. Předděličem kmitočtu získáme hodnotu 256 odpovídající maximální hodnotě 8 bitového čísla. Režim čítání ponecháme výchozí (Normal top=0xFF). Timer value, nebo-li počáteční hodnota čítače začíná v 0. Komparovaná hodnota je v tomto případě zbytek po děliči tedy FF hex (256 dec). Dosáhne-li komparátor zvolené úrovně, dojde k přerušení a odskočení do podprogramu řízení času hodin přesně každou sekundu, zároveň se komparátor vynuluje a



čítá znovu od počátku. Požadovali bychom např.: odskočení po 0,5s, hodnota komparované hodnoty by byla 128dec.

$$f_{d\dot{\epsilon}l} = \frac{f_{osc}}{128} = \frac{32768}{128} = 256Hz$$

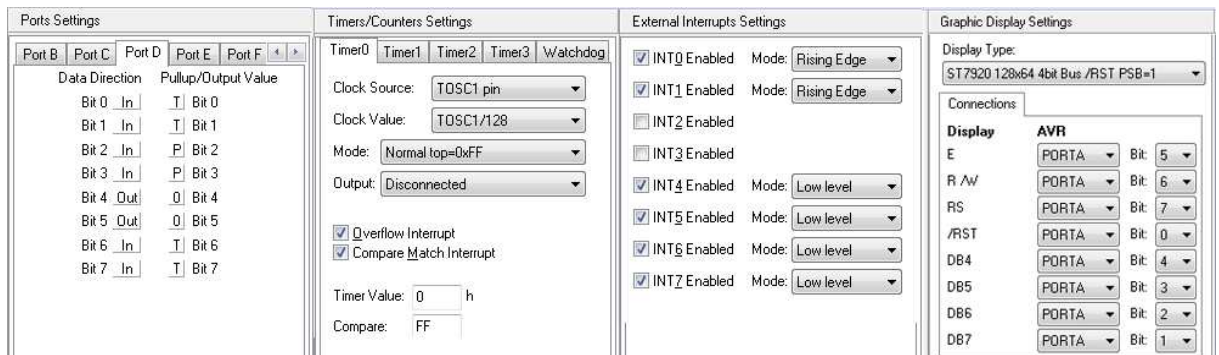
$$T_{d\dot{\epsilon}l} = \frac{1}{256} = 0,00390625s \quad (6)$$

$$T_{comp} = T_{d\dot{\epsilon}l} * Comp_{hod} = 0,00390625 * 256 = 1s$$

$$8b = 2^8dec = 256 dec = FF hex \quad (7)$$

K čítání a porovnání impulzů ze zpětné vazby motorů můžeme použít Timer1 a Timer2 na jehož vstupní piny T1 a T2 je přiveden signál fáze z krokového motoru posuvu a rotace. Pin by reagoval na vzrůstající hranu, porovnáním s druhou fází motoru by vyhodnotil směr otáčení a načítal počet impulzů. Nicméně frekvence generující motor při ručním posuvu nedosahují vysokých hodnot a lze tuto funkci obsloužit jednodušeji na vstupu vnějšího přerušení.

Poslední časovač Timer3 spravuje pravidelné odesílání frekvenčního signálu k modulům krokování. Řízení modulu vychází z maximální spočítané spínací frekvence 2500Hz pro motor posuvu vozíku. Z taktovacího oscilátoru MCU 16MHz odvozujeme požadovaný kmitočet přerušení. Z důvodu reakce modulů na vzestupnou hranu signálu a generování příznaku zapnuto/vypnuto musí být frekvence časovače Timer3 dvojnásobná, tedy 5000Hz k dosažení maximální vypočtené rychlosti. V podprogramu přerušení následně podmínkově ošetříme říditelnost spínání od 1Hz do plného rozsahu motoru 2500Hz. Rotace osy kamery pracuje v nižším rozsahu, obvykle desítkách hertz. Frekvence je z pravidla volena k dosažení půl otáčky těla kamery za časový interval přejetí celé dráhy, tak aby pokryla scénu před lineárním posuvem.



Obr. 45 – Okna počátečního nastavení  $\mu$ procesoru.

#### 4.1.4 Vnější přerušení

Vnější přerušení slouží pro okamžité odskočení programu do podprogramu a vykonání sledu činností s následným vrácením k odskoku z hlavního programu. Externí přerušení využíváme převážně u mechanických prvků. Najede-li vozík na koncový spínač, dojde k okamžitému zastavení motorů a vynulování kroků. Podobnou funkci zastávají i tlačítka start a stop uvádějící nastavený program pojezdu v činnost resp. zastavení. Uplatnit přerušení lze i pro kontrolu generovaných impulzů z komparátoru a jejich čítání, příp. porovnání. Reakce MCU může být povolena na nízkou logickou úroveň nebo na vzestupnou či sestupnou hranu.

##### Koncový spínač, strana volnoběžné kladky:

```

if(DIRECTION_MOVE == 0)           //kontrola příjezdu vozíku, ošetření směru příjezdu
{
    step_mov = 0;                  //vynulování čítacích kroků v Timeru3
    BUTTON_LED = 1;                //kontrolní rozsvícení LED
    DIRECTION_MOVE = 1;            //otočení směru posuvu,
                                    //tlačítko reaguje pouze na sestupnou hranu
    item = 2;                       //řízení grafické signalizace
    ETIMSK=0x00;                    //Vypnutí Timer0, zastavení stopek na displeji
    TCCR0=0x00;                     //Vypnutí Timer3, motory nedostávají pulzy
}                                    // k otáčení hřídele, klidový moment motoru zůstává

```

#### 4.1.5 Grafický displej

Nastavení grafické jednotky se zpřístupní po výběru typu řídicího integrovaného obvodu. Grafický displej s rozlišením 128\*64 a driverem ST7920 v paralelním zapojení komunikující v 4-bit režimu. Displej je připojen k bráně A s nadefinovanými porty dle schématu. Vstup RS v paralelním režimu označuje zápis instrukcí, či dat. RST je aktivní v log. 0. R/W přepíná čtení zápis. E port, povel ke čtení 4 bit sběrnice. Jednotka je napájena 5V. Kontrast znaků lze nastavit hardwarově SMD trimrem. Přenos dat sériový/paralelní je nezbytné zvolit propájením příslušného bodu označeného na DPS. Napěťovou úroveň podsvícení zobrazovače upravíme odporovým předřadníkem na hodnotu 3,3V.

Zabudovaný generátor předefinovaných alfanumerických znaků o formátu 8x16px na symbol nevyhovoval použití. K zobrazení menších znaků implementujeme knihovnu ASCII symbolů formátu 5x7px. Rozpoložení fontu umožní vytvořit textovou matici o 7 řádcích a 21 sloupcích. O vykreslování znaků a symbolů zvolené sady formátu se stará knihovna glcd.

Zahrnuje instrukce pro vkládání znaků jako textového řetězce, pohyb kurzoru po ploše, různé grafické úpravy ve formě odlišného rastru čar a kružnic.

## 4.2 Funkce programu

Program je rozdělen do tří hlavních částí. Foto režim, ukrývající další dva duplikáty pro možnost uchování vícero nastavení. Video režim, umožňující manuálně kontrolovaný přejezd z klávesnice a bluetooth mód zprostředkující ručně kontrolovaný přejezd vozíku pomocí joysticku.

### 4.2.1 Foto režim

Zastupuje primární činnost časosběrného snímání, čili snímání sekvenčního. Z tohoto důvodu je režim duplikován a rozdělen pro potřebu horizontálního a vertikálního posuvu.

Upravovanými parametry mohou být:

- Dráha snímání
- Snímkování
  - Snímky
  - Expozice
- Rychlost posuvu
- Rychlost rotace
- Směr rotace
- Odložení startu

Položka snímkování ukrývá rozšířené grafické submenu, s nastavením počtu snímků a délkou expozice. Mezi nimi jsou i parametry dopočítávané, vycházející z aktuálního nastavení. Perioda snímkování, odvozuje časový údaj o rozestupu jednotlivých snímků s potlačením délky expozice a položka celkový čas přejezdu kalkuluje s počtem snímků, délkou expozice a dobou přejezdu. Délka expozice stanovuje, jak dlouho bude vozík zastaven.

Tab. 12 – Ukázka výpočtu periody a celkového času.

Definované parametry	Dané parametry	
$krokování = 100Hz$ $počet\ snímků = 192$ $expozice = 1,2s$	$kroky\ motoru = \frac{360^\circ}{1,8^\circ} = 200\text{ kroků/otáčku}$ $S = \text{délka slideru} = 800\text{ mm}$ $O_{řemenice} = 2 * \pi * r = 119mm$	(8)

$\omega = \frac{f_{krok}}{200 * mikrokrok}$	$\omega = \frac{f_{krok}}{200 * 8} = \frac{100}{1600} = 0,0625 \text{ ot/s}$	(9)
$v = \omega * O_{\text{řemenice}}$	$v = 0,0625 * 119 = 7,4 \text{ mm/s}$	(10)
$t_{\text{přejezdu}} = \frac{s}{v}$	$t_{\text{pře.}} = \frac{800}{7,4} = 216 \text{ s}$	(11)
$T_{\text{snímkování}}$	$T_{\text{snim.}} = \frac{216}{192} = 1,125 \text{ s}$	(12)
$t_{\text{celkový}}$	$t_{\text{celkový}} = t_{\text{přejezdu}} + (\text{počet snímků} * \text{expoze})$	(13)
	$t_{\text{celkový}} = 216 + (192 * 1,2) = 6 \text{ min } 50\text{s}$	

Tab. 13 – Navrhovaná vizualizace grafického rozpoložení struktury menu. Tučně vyznačené položky označují vnořovací submenu.

Foto režim - přehled nastavení	rotace	V/H
Dráha	Vlast./vých. pozice	
<b>Snímkování</b>	x	
Rychlost posuvu	x m/s	
Rychlost rotace	x m/s	
Směr otáčení	< >	
Odložení startu	0:00:00	

#### 4.2.2 Video režim

Ovládá pouze manipulaci vozíku. Není zde prováděna synchronizace se spuštěním fotoaparátu. Snímání videa probíhá za chodu spínání motoru. Proto je zde velice důležitá plynulost přejezdu. Pulzace motoru by měla být minimální. Proto zvláště u vertikálního posuvu je nutné vybrat vhodnou přejezdovou rychlost, případně zařadit mikrokrokování ke spínání.

Tab. 14 – Navrhovaná vizualizace, video režimu.

Video režim/manuál	rotace	V/H
Dráha	Vlastní/výchozí pozice	
Směr posuvu	<-   ->	
Směr otáčení	<-   ->	
Rychlost posuvu	x,x	m/s
Rychlost rotace	x,x	m/s

### 4.2.3 Bluetooth režim

Bezdrátová kontrola pohybu vozíku usnadňuje kameramanovi práci s filmovou technikou. Během snímání jednou rukou koriguje rychlost posuvu joystickem a zároveň spolupracuje s fotoaparátem v těsné blízkosti. Kontroluje ostrost záběrů, kompozici, případně mění ohniskovou vzdálenost. Kameraman tak získává větší přehled o natáčené scéně a více prostoru ke korekci.



Obr. 46 – Ovladač s joystickem k bezdrátové kontrole manipulace.

## 4.3 Konfigurace motorů

První roztočení krokového motoru zvoleným driverem, zejména pak pohonu kontroly rotace fotoaparátu, bylo hrubé se znatelnými kroky. Z důvodů využívání rotace především v úhlu od 0 do 180° bylo nezbytné pohyb zpomalit. Řešením bylo rozdělení skoků na mikrokroky a získat větší rozlišení. S rozlišením vznikají vyšší spínací ztráty, klesá moment a vzrůstá nepříjemný hluk o vysokých frekvencích vlivem spínání. Nicméně jemnější pohyb nerozvibruje konstrukci. Experimentálním zkoušením režimů mikrokrokování byla zvolena hodnota velikosti kroku na 1/8. Toto krokování bylo nastaveno u obou driverů. S vyšším rozlišením by bylo nutné volit vyšší napětí zdroje z příčin vzrůstajícího odběru proudu projevující se nepříjemným „pískáním“ spínačů.

Provozované kmitočty jsou řádově od jednotek až k tisícům hertz. Dle charakteristiky krokového motoru viz obr. 47. Avšak motor rotace osy fotoaparátu pohybující blíže momentovému limitu bude provozován v pásmu do 100 Hz.

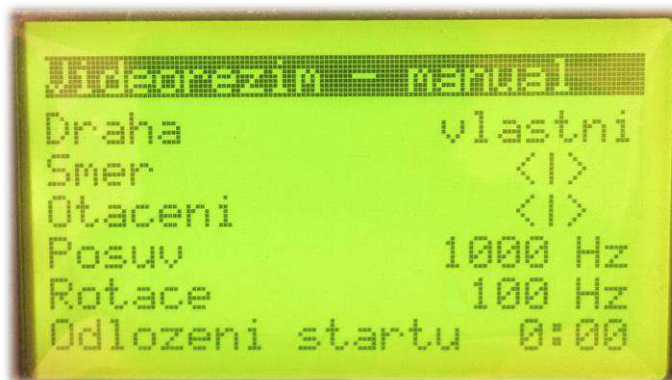
#### 4.4 Struktura programu

Tělo programu je složeno z jednotlivých položek „case“ FOTO, VIDEO, BLUETOOTH, SNIMKOVANI, TIMELAPSE a NATACENI. Princip přístupu vychází ze stavového automatu. Blokově jsou rozděleny na obsluhu grafického rozhraní a logického řešení funkcí tlačítek s poměny. Grafické rozhraní vyžadovalo nastavit vhodné rozestupy mezi symboly, zarovnat text, kurzorovou značku aktuálně vybrané nabídky, přepisování textové nabídky po změně parametrů a souřadnicové umístění vypisovaných proměnných. Funkční blok obsahuje čtení stavu tlačítek s následnou vazbou na proměně využívané k definování parametrů, matematické operace, přeskokování mezi jednotlivými nabídkami s provázáním na přerušení.

#### 4.5 Ovládání zařízení

Ovládání zařízení je specifické v přístupu ze dvou stran (levá i pravá strana kolejí), obsluha je vždy za fotoaparátem. K tomu se rovněž upravila uživatelská ovladatelnost. Pro snadné čtení a obsluhu tlačítek jsou popisky zdvojeny s opačnou orientací. Menu nabídku displeje lze otočit o 180°. Polohu přístupu tak nastavujeme tlačítkem AUTO, měnicí funkce tlačítek a zobrazení displeje. MENU tlačítkem přepínáme mezi jednotlivými režimy foto, video a bluetooth. Spínače LEFT a RIGHT spustí příkaz k přejezdu vozíku do výchozí polohy kraje kolejiště. SELECT slouží pro deaktivaci H-můstku driveru TB6560 a volnému pohybu vozíku. Při dlouhém stisku dojde k snímání pulzů ze zpětné vazby motoru. Tlačítko ON/OFF je funkčně podobné. Vyřadí z činnosti driver DRV8825 pro možnost volné rotace a načítání impulzů z pohonné jednotky. Pro společné načítání impulzů v době odpojení driverů stiskneme tlačítko START.

Druhá strana krabičky je osazena kurzory pro pohyb v menu, definování parametrů přejezdu, snímání atd. Poslední tlačítka START a STOP slouží pro uvedení nastavených veličin v chod resp. k jejich zastavení a přerušení.



Obr. 48 – Nabídka displeje, video režim.

## 5 Cenová kalkulace

Většina mechanických dílů či elektronických součástek byla objednána v zahraničním asijském internetovém obchodě. S přihlédnutím k dlouhé dodací lhůtě byla objednávka kritických částí zdvojnásobena pro případ poškození během transportu nebo vlivem nevhodného zacházení, milným zapojením či jinými možnými komplikacemi. Některé mechanické prvky v blízkých rozměrových škálách byly objednány po skupinách. Zahraniční nabídka komponentů je ekonomicky velmi výhodná. Zakoupené hotové elektronické moduly nepřevyšovaly cenu hlavního integrovaného obvodu nabízeného v kamenných prodejnách v ČR. Nákupní cena se vždy vešla pod stanovenou hranici platby DPH (krom hlavní konstrukce).

Tab. 15 – Kalkulace položek posuvu

Mechanická část			Elektronická část		
Položka	kusů	cena	Položka	kusů	cena
Ashanks lineární posuv	1+DPH	3932	DRV8825	3	80
Řemenice	5	258	TB6560	1	110
Řemen	1m	100	23HS5628	1	418
Whitworthovy šrouby	10	88	17HS2408	1	154
Upínací destička	1	198	Bluetooth HC - 05	1	60
			ATmega128A	10	192
			SMD k osazení	x	300
			Kabeláž	3	110
			Ovladač	1	55
			Klávesy	4	136
			Display 128*64	2	220
			LiIon balancer	1	42
			Koncové spínače	10	40
<b>Celková suma</b>		<b>6493 Kč</b>			

## Závěr

Sestavení kompletního výrobku s odladěním veškerých chyb, s kterými se konstruktér, návrhář, elektronik a programátor potýká, je velice složité. Řešení problémů vznikajících během stavby se s odstupem času můžeme dívat odlišněji. Navrhování, sestavování je o neustálém cyklu zlepšování či updatu firmwaru řízení s intuitivním uživatelským rozhraním. Proto za konečnou verzi produktu stojí mnoho úsilí a snahy.

Zhodnotím-li navrhovanou-modifikovanou konstrukci, musím ocenit kompaktnost celku a implementaci veškerých modulů do těla konstrukce či vnější umístění ve formě bočních „batůžků“ uchycených na kolejových příčnicích. Celá konstrukce je velice mobilní, vozík lze snadno odsunout na dané místo, polohově zajistit opěrným šroubem a pohodlně celou dlaní obejmou karbonovou kolejnici k přenosu zařízení. Motorová modifikace se vyvedla zejména díky dnešní dostupnosti 3d tisku, obecně velkým pomocníkem pro jakoukoli jednocílovou výrobu. Jednotlivé pomocné díly byly navrhovány s maximální efektivností pro využití předností konstrukce a hotových závitových otvorů. Tím se minimalizovala složitost zabudování nových dílů do konstrukce. Přesné účelové modelování však vyžadovalo značnou trpělivost měřením vztyčných bodů na konstrukci (milimetry až jednotky desetin mm). Teplotní stálost všech 3d výtisků (PLA plastů) je do cca 60°C. Proto je nezbytné nezatěžovat motory do uváděných krajních hodnot, kdy teplota motoru může dosahovat maximálně 80°C. Rozpoložení všech hmotných částí je směřováno do osy kolejiště. Pouze stranová nevyváženost kompenzována alespoň z části hmotností baterie je drobným problémem. Možným řešením je vložení 8-mi článkové baterie (nyní 6 článků) a změnit krokový motor s přírubou NEMA23 za lehčí ekvivalent s nižší hmotností potažmo momentem. Celek lze umístit na zem, pevnou podložku nebo stativ. Zkouška na stativu odhalila drobné komplikace, kterým je třeba se vyvarovat.

Snímáme-li ve „vertikální“ poloze 45°- 90° udržení kolejiště stativovou hlavou není problematické. Kolejiště se opírá o zem a menší úhel nelze nastavit kvůli dispozici stativu. Motor vozík udrží, nicméně je nutné v této poloze značně nahnout řemen, aby nedocházelo k jeho prokluzu na řemenici.



Horizontální přejezd lze provozovat bez stativové hlavy a stativ upnout přímo k destičce. (Kulová stativová hlava nedovede unést stranové nevyvážení a dochází k samovolnému klesání slideru). K přejezdu lze využít celou délku dráhy bez omezení, jen vyrovnání horizontu může být komplikací.

Jiné úhlové nastavení pozice kolejíště je možné sestavit s doplněním druhého stativu nebo v lehčí variantě monopodem. Monopod je pouze podpěrná „noha“ kompenzující zatížení pojezdu v krajní poloze, nedrží však stabilitu.

Řídící elektronika je dimenzována dostatečně s pokrytím potřeb k řízení činnosti časosběru, manuálního přejezdu i bezdrátové komunikace. Rozmístění elektroniky můžeme rozdělit na hlavní a předsunuté. Hlavní řídící elektronika je umístěna spolu se spínacím modulem činnosti posuvu. Navrhované DPS s MCU mohlo dosahovat větší integrity součástek a rozměrově zmenšit plochu desky. K dramatické redukci rozměrů krabičky by však nedošlo, protože musela být zachována délka pro umístění grafického displeje do víčka. Z vyplívající rozměrové problematiky pro další generace je vhodné přenést zobrazovací jednotku do mobilního telefonu. Tímto počinem by odpadly i veškeré (špatně těsnící) ovládací prvky a elektronické vybavení by mohlo být umístěno v menší modulu odolávající vlhkosti i vodě a dosáhnout vyššího stupně krytí IP. Modul TB6560 by mohl být rovněž vyměněn za podobně malý modul jako DRV8825, ale dosahující vyšších proudů. Konkrétně za modul MP6500. Nicméně s aktuálním modulem TB6560 není problém, spínací proud je technicky možné částečně řídit vnějšími optočleny a spínání cívek je neslyšitelné.

Předsunutá elektronika ukryta pod vozíkem v domečku motoru je driver rotace fotoaparátu DRV8825. Do dutiny je přiveden svazek komunikačních a napájecích kabelů. Svazek je poté veden k řídící elektronice. Tato problematika tažení kabelů je zde vyřešena a nikterak neomezuje v použití. Bezdrátový přenos by však byl zajímavou alternativou, pokud by moduly našly místo ve spodní ploše vozíku. DRV8825 vydává slyšitelnou frekvenci od spínání. Lze ji částečně potlačit vyšším napětím.

Zpětná vazba motorů k načítání pulzů pro odečtení polohy není funkční dle původních představ a simulace. Generované pulzy mohou být z části utlumeny na H-můstku driveru nebo mechanická zátěž hřídele nedovolí k dynamickému skoku rotoru pro vyvolání potřebné

úrovně pulzu. Motor posuvu vozíku lze stále doplnit optickým kotoučem pro odečet rotace hřídele. Nicméně druhý motor zakrytý v těle vozíku je nedostupný.

K sekvenčnímu časoběrnému snímání jsem plánoval i úpravu techniky fotoaparátu, spíše objektivů. Snímání stovek až tisíců obrázků se negativně nejvíce podepisuje na mechanických částech fotoaparátu. V prvním případě trpí zrcátko fotoaparátu při sklápění a propouštění světla k snímači. Tento efekt však eliminují bezzrcadlové fotoaparáty tzv. Single-Lens Translucent. Místo zrcátka disponují pevným hranolem dělící světlo pro snímač a senzory. Druhým mechanicky namáhaným členem je clona objektivu. Clona je vždy plně otevřena a k zaclonění dojde až těsně před pořízením snímku. K vyřazení je třeba fotografovat s plně otevřenou clonou a zaclonění provádět ručně. Vložení upraveného adaptéru mezi „pacičky“ fotoaparátu a objektivu navýšíme uměle clonové číslo.

### Výsledné parametry zařízení:

<b>Hmotnost</b>		<b>Rozměry</b>	
		<b>DxŠ</b>	
Lineární posuv	3,1 kg	Celá konstrukce	1155 x 180 mm
Baterie	0,44 kg	Vozík	120 x 150 mm
Snímací technika	0,95 kg	Délka kolejí	920 mm
Stativová kulová hlava	0,4 kg	Manipulační dráha	800 mm
Celková hmotnost	4,89 kg	Rozteč kolejí	90 mm

### **Provozní parametry**

Výdrž baterie	5h dle provozu
Max. rychlost přejezdu	185,9 mm/s
Min. rychlost přejezdu	- mm/s
Kabelová synchronizace	dvoustavová
Mikrokrokování	1/8 kroku
Osová rotace kamery	360 °

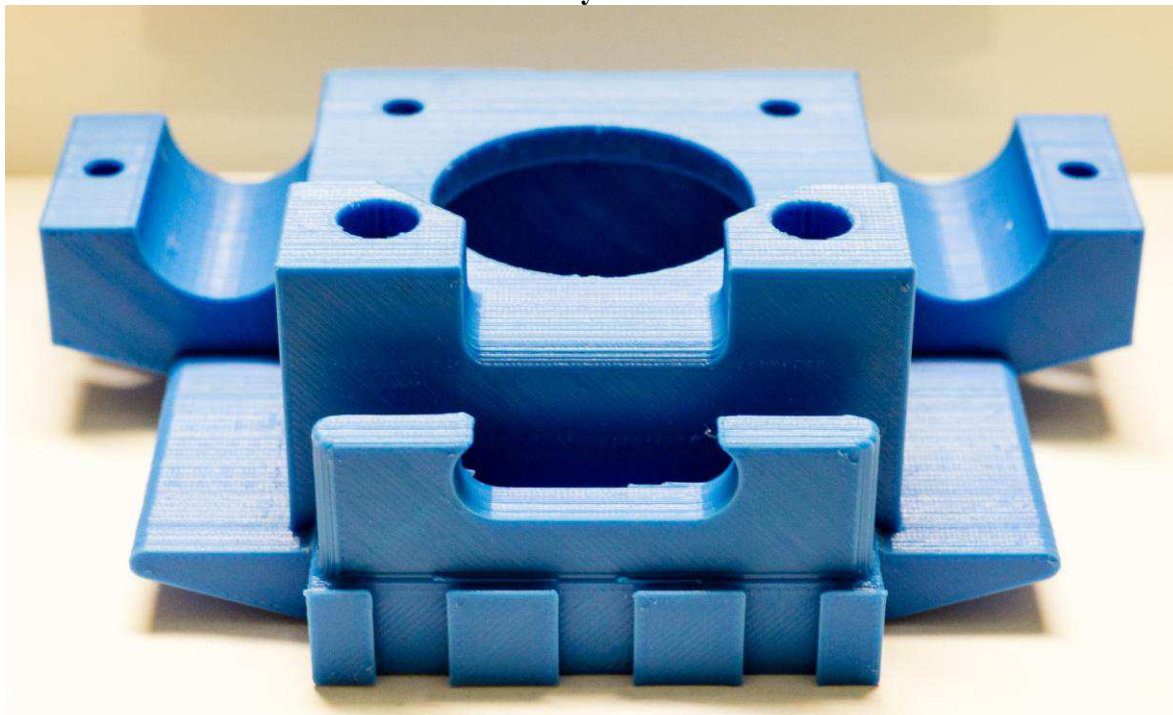
## Seznam literatury

- [1] Uhlíková vlákna. Duratec [online]. [cit. 2018-04-11]. Dostupné z: <http://www.duratec.cz/cs/techinfo/pouzivane-materialy/uhlikova-vlakna-karbonove-kompozity/>
- [2] Carbon Fiber: All Patterns Explained. Carbon.ee [online]. 26.04.2015 [cit. 2018-04-11]. Dostupné z: <http://www.carbon.ee/en/n/carbon-fiber-all-patterns-explained>
- [3] DSLR camera linear slider: ASHANKS 100cm. ASHANKS [online]. [cit. 2018-04-11]. Dostupné z: <http://www.ashanks.com/track-slider-with-adjustable-angle-tube-follow-focus-pan-ashanks-video-photographic-equipment-p659207.html>.
- [4] 3D TISK, PRINCIPY, TECHNOLOGIE. ABC 3D [online]. [cit. 2018-04-11]. Dostupné z: <https://www.abc3d.cz/o-3d-tisku-neprehlednete/115-3d-tisk-principy-technologie>
- [5] Základní rozdělení: 3D tiskárny FDM. Makerslab [online]. [cit. 2018-04-11]. Dostupné z: <https://www.makerslab.cz/typy-3d-tiskaren/>
- [6] VOCHOMŮRKA, Jan. Typy trysek. Openinnovations [online]. 14.08.2017 [cit. 2018-04-11]. Dostupné z: <http://www.makerslab.cz/blog/2017/08/typy-trysek/>
- [7] FELIX 3.0. Felixprinters [online]. [cit. 2018-04-11]. Dostupné z: <http://www.felixprinters.cz/popis-3d-tiskarny/>
- [8] Parts that make up a hotend, and what do they do. In: 3dprinting [online]. [cit. 2018-04-25]. Dostupné z: <https://3dprinting.stackexchange.com/questions/1519/what-are-the-parts-that-make-up-a-hotend-and-what-do-they-do>
- [9] MATERIÁLOVÝ SLOVNÍK. Material pro 3d [online]. [cit. 2018-04-11]. Dostupné z: <https://www.materialpro3d.cz/materialovy-slovník/>
- [10] Rozdiel medzi ABS a PLA. Tvar 3D [online]. 28.11.2013 [cit. 2018-04-11]. Dostupné z: <http://tvaroch.sk/blog/rozdiel-medzi-abs-pla/>
- [11] VOCHOMŮRKA, Jan. Data pro 3D tisk: STL - file. MakersLab [online]. 26.05.2016 [cit. 2018-04-11]. Dostupné z: <https://www.openinnovations.cz/blog/2016/05/data-pro-3d-tisk/>
- [12] JECH, Adam. Příručka pro designéry: Základní pravidla. MakersLab [online]. 17.09.2015 [cit. 2018-04-11]. Dostupné z: <https://www.makerslab.cz/blog/2015/09/prirucka-pro-designery/>
- [13] STL (file format). In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2018-04-26]. Dostupné z: [https://en.wikipedia.org/wiki/STL\\_\(file\\_format\)](https://en.wikipedia.org/wiki/STL_(file_format))
- [14] KOVÁČIK, Miroslav. NASTAVENÍ KVALITY 3D TISKU. 3dfactory [online]. Prosinec 25, 2017 [cit. 2018-04-26]. Dostupné z: <https://3dfactory.cz/2017/12/25/nastaveni-kvalita-3d-tisk/>
- [15] Step-Up/Down/ Inverting Switching Regulators: datasheet. Sparkfun [online]. 2003 [cit. 2018-05-01]. Dostupné z: <https://www.sparkfun.com/datasheets/IC/MC34063A.pdf>
- [16] Atmel: ATmega128A - datasheet. Microchip [online]. 09/2015 [cit. 2018-05-01].

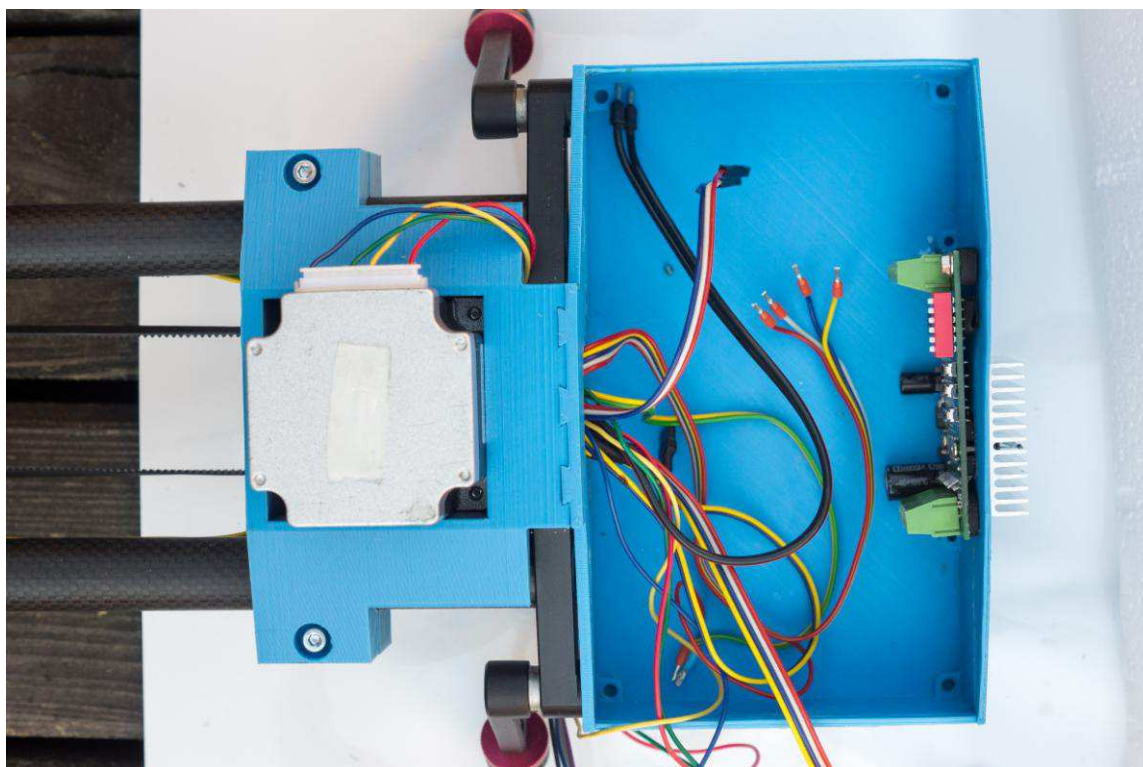
- Dostupné z: [http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-8151-8-bit-AVR-ATmega128A\\_Datasheet.pdf](http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-8151-8-bit-AVR-ATmega128A_Datasheet.pdf)
- [17] Vlášek, Josef. Řídící elektronika krokového motoru. V Plzni, 2016. Bakalářská práce (Bc.). Západočeská univerzita v Plzni, Fakulta elektrotechnická. Vedoucí práce Kamil Kosturik.
- [18] DRV8825 Stepper Motor Controller IC. Texas instruments [online]. April 2010 [cit. 2018-05-01]. Dostupné z: <https://www.pololu.com/file/0J590/drv8825.pdf>
- [19] TOSHIBA BiCD Integrated Circuit: TB6560AHQ. Toshiba [online]. 2014-10-01 [cit. 2018-05-02]. Dostupné z: <https://www.toshiba.semicon-storage.com>
- [20] Stepper Motor Driver: Single Axis (Peak 3A) TB6560. *Online trading* [online]. [cit. 2018-05-02]. Dostupné z: <http://qqtrading.com.my/stepper-motor-driver-single-axis-3amp-tb6560>
- [21] 31306-MS: Optocouplers 3A, FULL BRIDGE STEPMOTOR DRIVER. *Marlin P. Jones & Assoc. Inc.* [online]. [cit. 2018-05-02]. Dostupné z: <http://www.mpja.com/download/31306ms.pdf>
- [22] DRV8825 Stepper Motor Driver. *Pololu* [online]. [cit. 2018-05-02]. Dostupné z: <https://www.pololu.com/product/2132>
- [23] HC Serial Bluetooth Products: User Instructional Manual. *Postavrobota.cz* [online]. [cit. 2018-05-03]. Dostupné z: [https://www.postavrobota.cz/fotky46704/fotov/\\_ps\\_1hc\\_hc-05-user-instructions-bluetooth.pdf](https://www.postavrobota.cz/fotky46704/fotov/_ps_1hc_hc-05-user-instructions-bluetooth.pdf)
- [24] HC - 05: Bluetooth to Serial Port Module. *Electronica estudio* [online]. [cit. 2018-05-03]. Dostupné z: <http://www.electronicaestudio.com/docs/istd016A.pdf>
- [25] 57HS09: krokový 2-fázový motor 1,3Nm. *Cncshop.cz* [online]. [cit. 2018-05-03]. Dostupné z: <http://www.cncshop.cz/57hs09-krokovy-2-fazovy-motor-1-3nm>
- [26] Nema23 23HS5628 Stepper Motor. *Aliexpress* [online]. [cit. 2018-05-03]. Dostupné z: [https://www.aliexpress.com/item/Free-shipping-4-lead-Nema-23-23HS5628-Stepper-Motor-57-motor-165-Oz-in-56mm-2/32828834665.html?aff\\_platform=aaf&cpt=1525342733387&sk=jiEmIUR&aff\\_trace\\_key=d76f404875e149be98e274ad57c33afc-1525342733387-03548-jiEmIUR&terminal\\_id=8506b741718441eea4a1fdc85fa2ff70#nm](https://www.aliexpress.com/item/Free-shipping-4-lead-Nema-23-23HS5628-Stepper-Motor-57-motor-165-Oz-in-56mm-2/32828834665.html?aff_platform=aaf&cpt=1525342733387&sk=jiEmIUR&aff_trace_key=d76f404875e149be98e274ad57c33afc-1525342733387-03548-jiEmIUR&terminal_id=8506b741718441eea4a1fdc85fa2ff70#nm)
- [27] Nema23 23HS5628 Stepper Motor. *Aliexpress* [online]. [cit. 2018-05-03]. Dostupné z: [https://www.aliexpress.com/item/Free-shipping-4-lead-Nema-23-23HS5628-Stepper-Motor-57-motor-165-Oz-in-56mm-2/32828834665.html?aff\\_platform=aaf&cpt=1525342733387&sk=jiEmIUR&aff\\_trace\\_key=d76f404875e149be98e274ad57c33afc-1525342733387-03548-jiEmIUR&terminal\\_id=8506b741718441eea4a1fdc85fa2ff70#nm](https://www.aliexpress.com/item/Free-shipping-4-lead-Nema-23-23HS5628-Stepper-Motor-57-motor-165-Oz-in-56mm-2/32828834665.html?aff_platform=aaf&cpt=1525342733387&sk=jiEmIUR&aff_trace_key=d76f404875e149be98e274ad57c33afc-1525342733387-03548-jiEmIUR&terminal_id=8506b741718441eea4a1fdc85fa2ff70#nm)
- [28] BELZA, Josef. Akumulátory Li-ion a jejich nabíjení. *Belza.cz* [online]. 24. 3. 2001 [cit. 2018-05-07]. Dostupné z: <http://www.belza.cz/charge/liion1.htm>
- [29] Capteur de position linéaire. *Directindustry.fr* [online]. [cit. 2018-05-07]. Dostupné z: <http://www.directindustry.fr/prod/ams-ag/product-27174-1457983.html>
- [30] Epilog Laser System Maintenance. *Factory enova* [online]. June 12, 2014 [cit. 2018-05-07]. Dostupné z: <http://factoryenova.com/blog/2014/06/12/epilog-laser-system-maintenance/>
- [31] راه اندازی ان کوردر موتور Novin Tarasheh [online]. June 12, 2014 [cit. 2018-05-07]. Dostupné z: <http://ntaco.ir/2-وبلاگ/page/4/>
- [32] Stepper motor driver. *Hteck.ca* [online]. [cit. 2018-05-07]. Dostupné z: <http://www.hteck.ca/electronics/stepper-motor-drv/sm-driver.html>
- [33] AVR910: In-System Programming [online]. November, 2000 [cit. 2018-05-09]. Dostupné z: <https://www.elnec.com/sw/avr910.pdf>

## Přílohy

### Příloha A – Fotodokumentace konstrukce výrobku

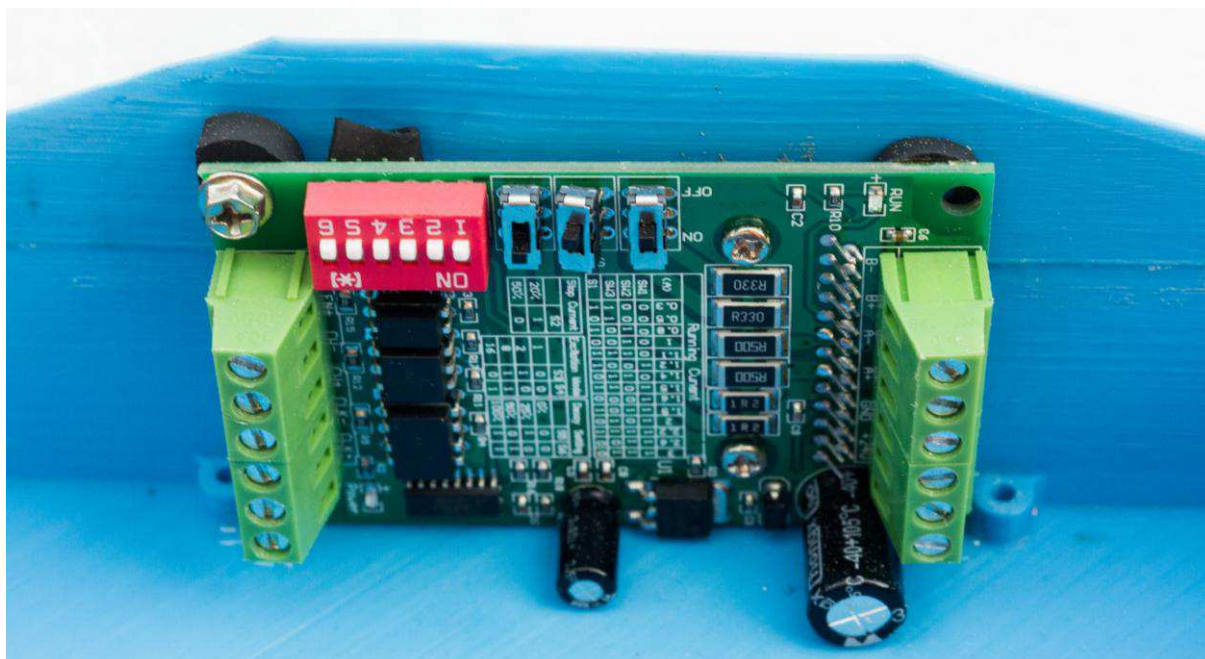


Obr. A1 – Vytisknutý model domečku krokového motoru posuvu, uložen z části na kolejkách a příčnicku. Pohled zdola.

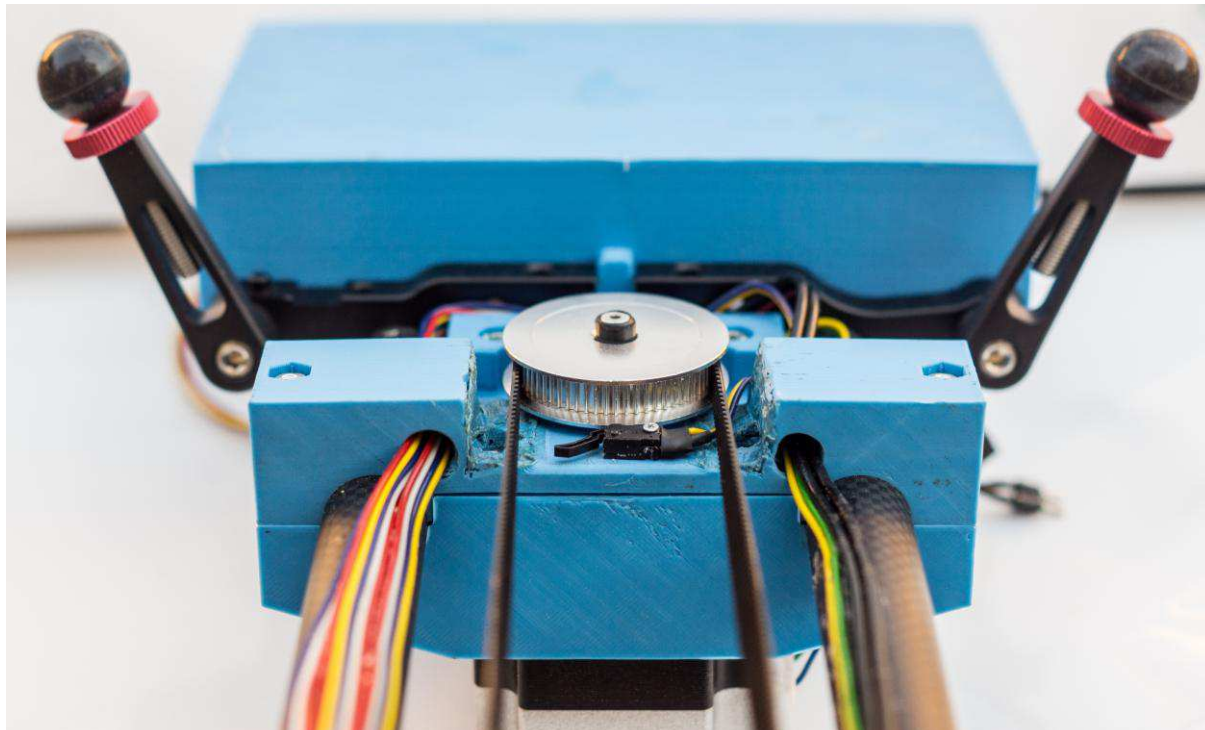


Obr. A2 – Spojení vytisknutých modelů mechanickým zámkem a šrouby v montážních otvorech k příčnicku. Pod kolejkami je zavlečena kabeláž vyústěna v krabici elektroniky. Driver krokového motoru je implementován ve stěně s chlazením mimo vnitřní prostor.

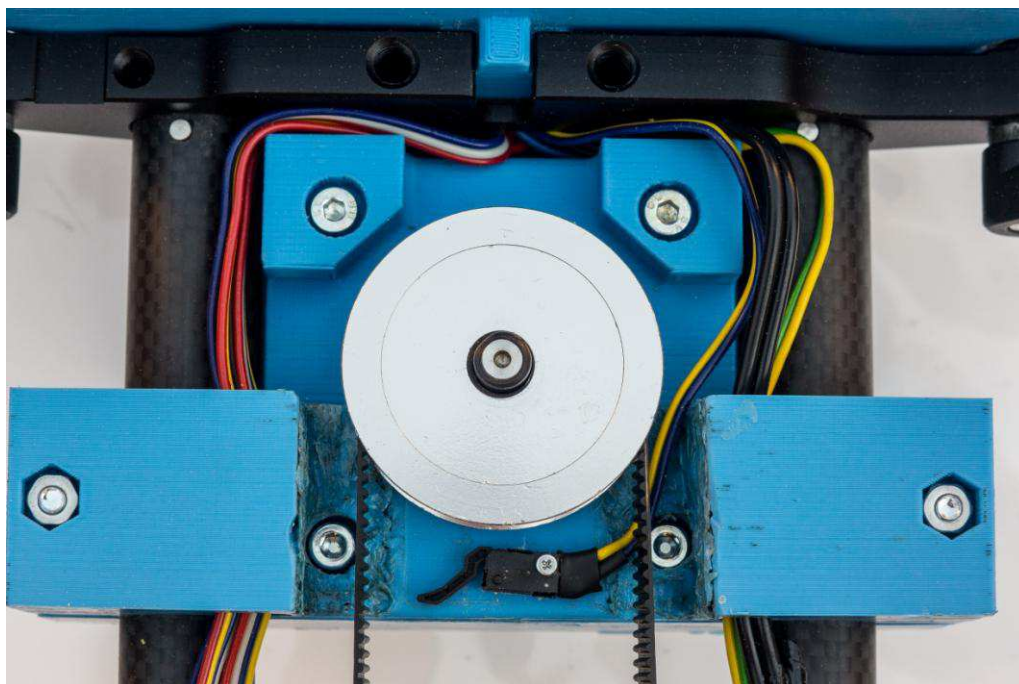




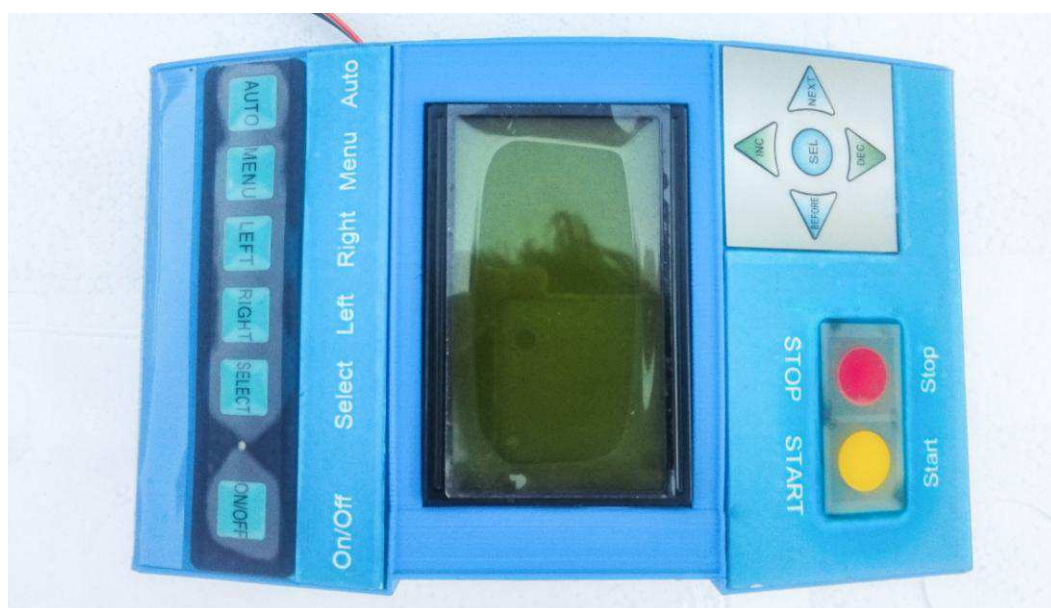
Obr. A3 – Pohled na uložení driveru. Chlazený integrovaný obvod je umístěn z druhé strany šroubově spojený skrz otvor k chladiči. Přepínače slouží k nastavení parametrů driveru. Vodorovná nesymetrie ve stěně krabičky vznikla nedopatřením při 3D tisku. Horní část obvodu krabičky je lepena dodatečně k sobě.



Obr. A4 – Obvodové kabelové vedení po spodní části kolejí nebrání v úchopu a manipulaci.

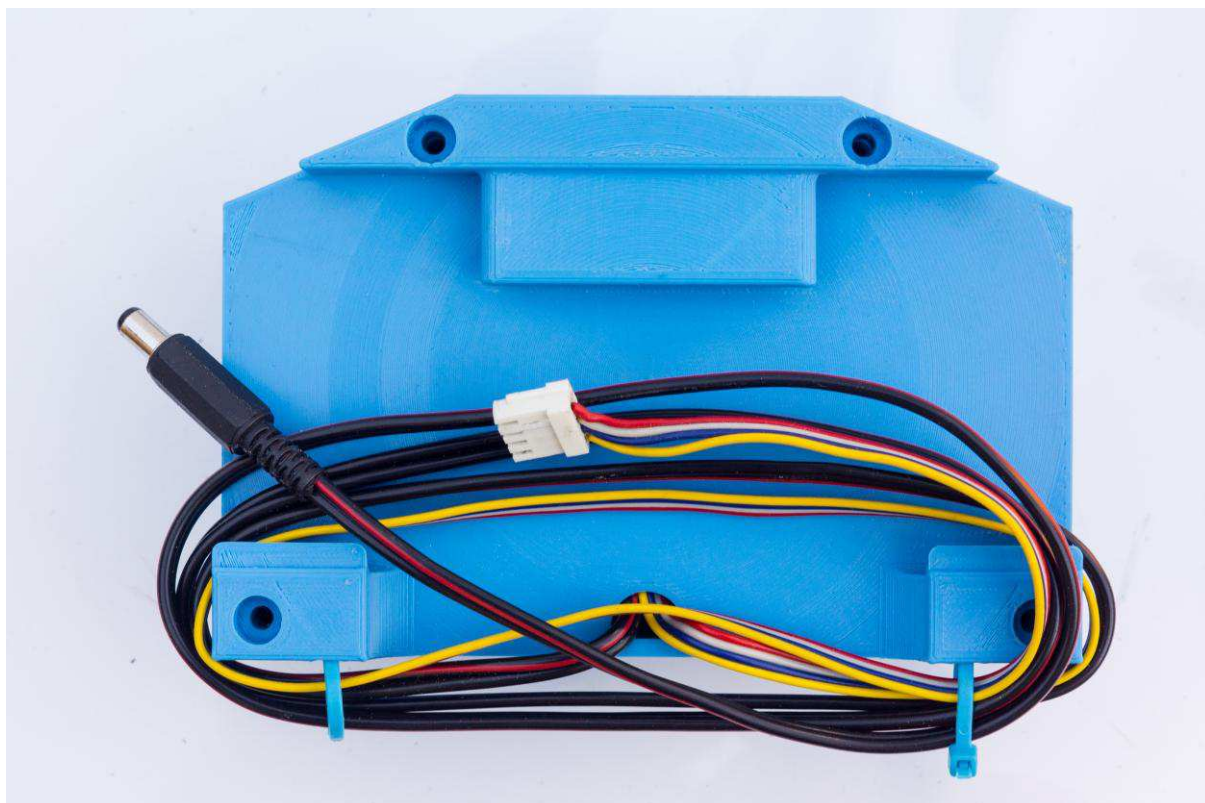


Obr. A5 – Z důvodu většího průměru řemenice muselo dojít k úpravě spodního dílu. Mezi řemenem je umístěn koncový spínač.

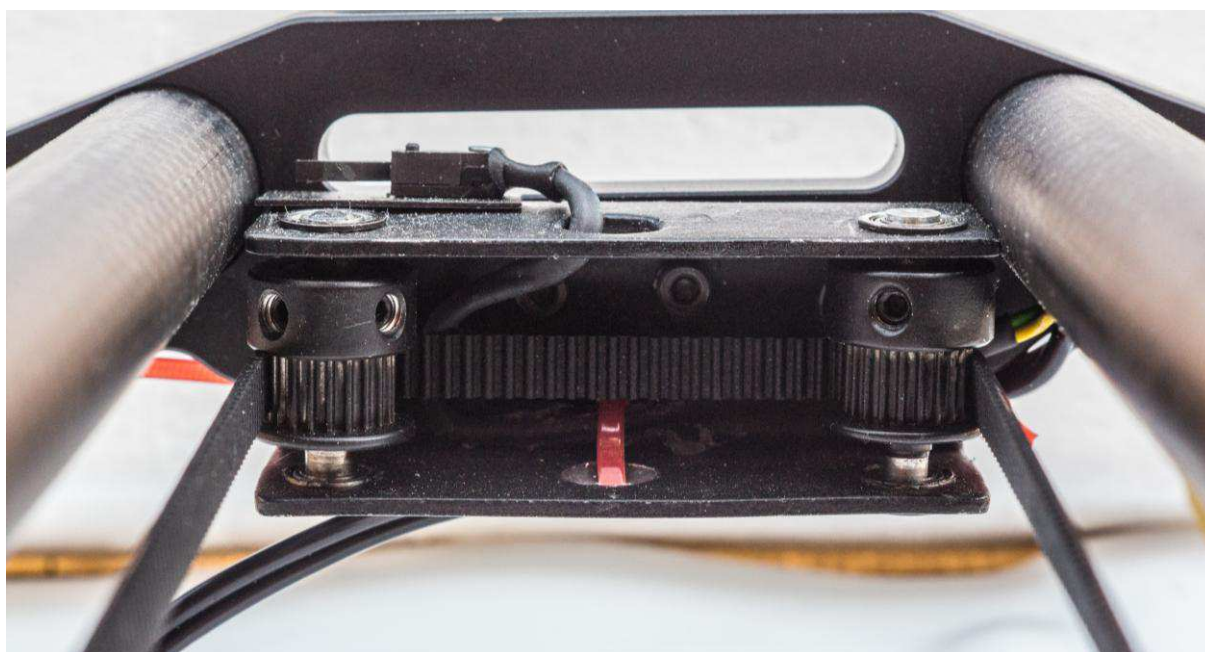


Obr. A6 – Víčko krabičky s ovládacími prvky a zobrazovací jednotkou. Možnost přístupu k jednotce je z pravé i levé strany.



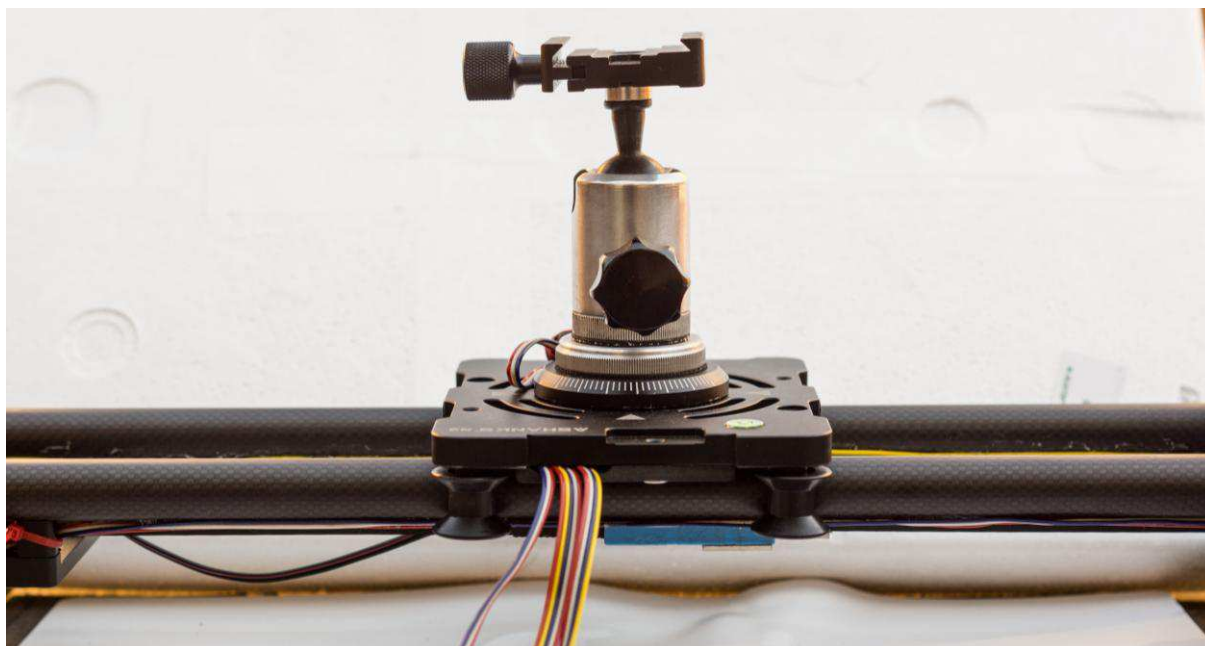


Obr. A7 – Battery pack s úchytem na příčnick, protizávaží motoru. Snadno přenosný. Napájecí konektor doplněn vývody potenciálů článků pro balancér dobíjecí stanice.

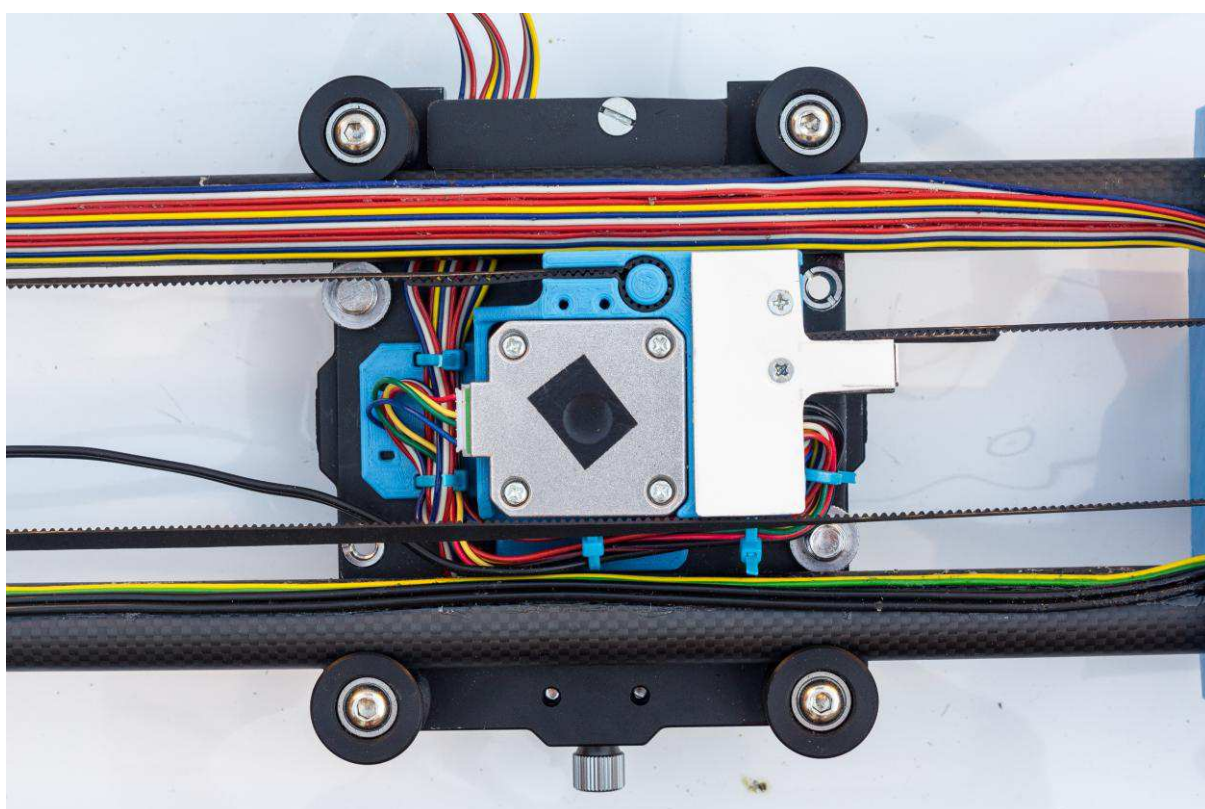


Obr. A8 – Můstek volnoběžné kladky s dvěma řemeničkami. Koncový spínač.



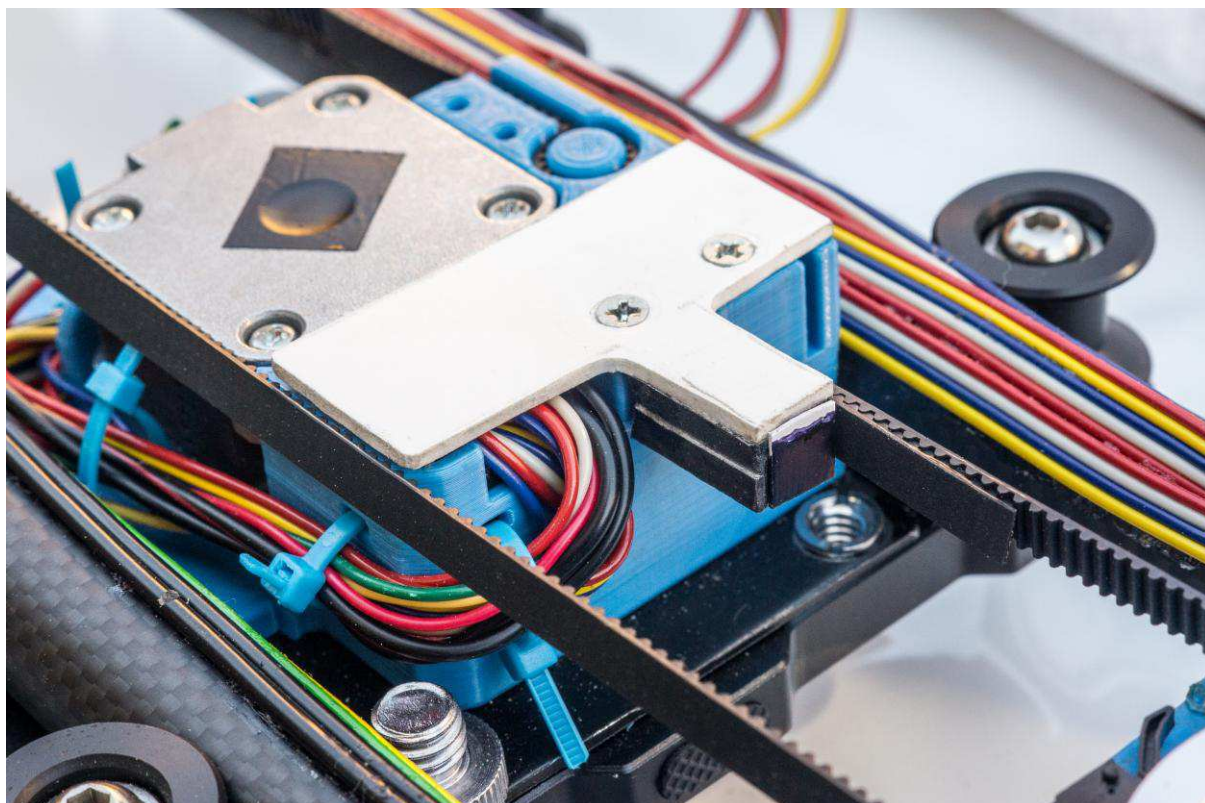


Obr. A9 – Vozík s namontovanou stativovou hlavou. Vpředu přívod kabeláže pro spoušť a ovládání motoru.



Obr. A10 – Spodní část vozíku s modulem domečku krokového motoru s připravenými řemenovými zámky, otvory pro stahovací pásky kabeláže a kapsou pro spínací driver motoru (zakryto bílou destičkou s ramenem pro stisk koncového spínače).





Obr. A11 – Detail vstupu kabeláže k driveru, řemene procházejícího těsně kolem motorové základny a dojezdového ramene vyměřeného ke koncovému spínači.



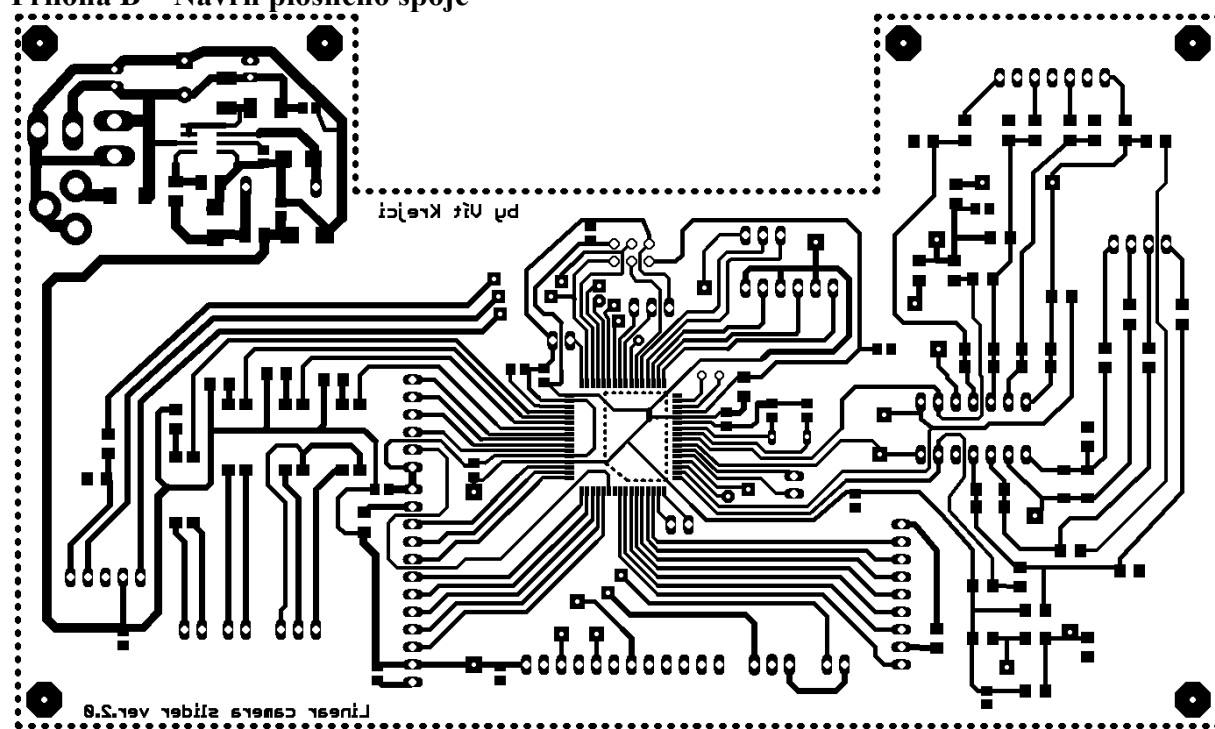
Obr. A12 – Horizontální rozložení stativu bez kulové hlavy. Stabilní v celé délce pojezdu.



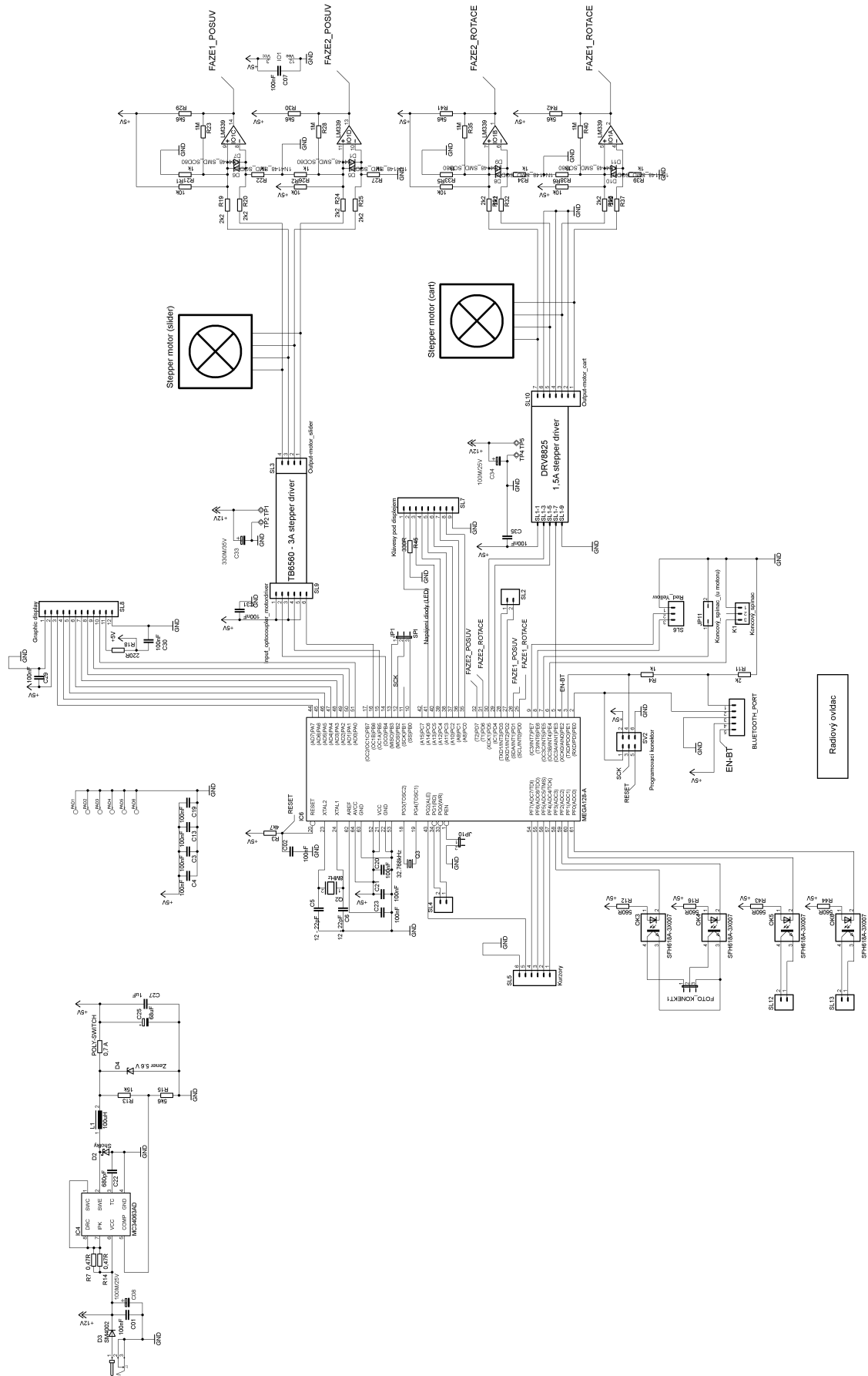


Obr. A13 – Vertikální rozložení stativu v úhlu 45° – 90°. Tažný řemen je nutné dostatečně napnout.

Příloha B – Návrh plošného spoje



Obr. B1 – Deska plošného spoje spodní pohled, bez polygonu „rozlité“ mědi potenciálu GND.



Obr. B2 – Schéma řídicí elektroniky.

**Příloha C – Řídící program**

```

/*****
Project : Linear camera slider
Version : 2.0
Date   : 17.1.2018
Author  : Vít Krejčí

Chip type      : ATmega128A
Program type   : Application
AVR Core Clock frequency: 16,000000 MHz
Memory model   : Small
External RAM size : 0
Data Stack size : 1024
*****/
#include <mega128a.h>
#include <delay.h>

// Graphic Display functions
#include <glcd.h>

// Font used for displaying text
// on the graphic display
#include <font5x7.h>

// Declare your global variables here
#define FOTO          0
#define VIDEO         1
#define BLUETOOTH     2
#define SNIMKOVANI    3
#define TIMELAPSE     4
#define NATACENI      5

unsigned char menu = 0;
unsigned char string[] = {'0', '0', '\0'};

//Odečet času - timer0
unsigned long int time = 0;
unsigned long int time_ms = 0;
unsigned long int last_delta_time = 0;
unsigned long int last_time = 0;
unsigned char H = 0, M = 0, S = 0;

bool uvod = 1;
unsigned int step_mov = 0;
unsigned int step_rot = 0;
bool fast_step_move = 0;
bool prerus = 1;
unsigned int def_step_mov = 100;
unsigned int def_step_rot = 600;
//unsigned long int sec;

unsigned char position = 12;

//proměné pro výpočty
unsigned char pull = 119;
unsigned int picture = 50;
long int time_move;

int total_time;

//Inicializace při změně menu nabádky
//Čítání pŕlžů časovače pro porovnání posuvu
//Čítání pŕlžů časovače pro porovnání rotace
//maximální rychlost posuvu
//flag podmínka pro posuv
//Výchozí hodnota 2500=hodnota časovače 1Hz
//Výchozí hodnota 2500=hodnota časovače 1Hz

//pozice kurzoru

//2*pi*r řemenice

//800(((2500-def.step)/1600)*119) 800 delka pojezdu, 2500
//perioda citace, 1600 počet kroků k otočce motoru, 119 řemenice
// time_move+(picture*expozice)

```

```

int per_of_shot; //perioda focení
unsigned char step_of_shot = 12; // zaokrouhlení ?
unsigned char per_step = 0;
int expozice;
//unsigned int expo_pom = 0; //časová perioda 0,0002 s

unsigned char item = 1; //položka zobrazení směru

void numToStr(unsigned char num, unsigned char* str);

//Koncové spínače
#define LIMIT_SWITCH_START PINE.5 // koncový spínač 1 (u motoru)
#define LIMIT_SWITCH_END PINE.4 // koncový spínač 2 (u baterie)

//Optočlen - spínáno přes optočleny, přepnutím do nuly --> optočlen sepne
#define SHOT PORTF.3 // ostření a fixace expozice
#define FOCUS PORTF.2 // ovládání závěrky (vytvoření snímku)
#define CURRENT_LINMOTOR PORTF.1 // NASTAVENÍ LIMITUJÍCÍHO PROUDU
// VINUTÍ MOTORU (PRO FÁZI)
#define STOPCURRENT_LINMOTOR PORTF.0 // Proud fází vynutí, při stání motoru

//Kurzory
#define BUTTON_UP (PINF.6 == 0)
#define BUTTON_DOWN (PINF.7 == 0)
#define BUTTON_NEXT (PINF.4 == 0)
#define BUTTON_BEFORE (PINF.5 == 0)
#define BUTTON_ENTER PING.2 // if(PING.2 == 1) -> if(PING & 0x04 == 1) -->
// POKUD JE NA VSTUPU 1 - SPLNĚNO

//Tlačítka
#define BUTTON_AUTO_ROTATION (PINC.0 == 0)
#define BUTTON_MENU (PINC.1 == 0)
#define BUTTON_LEFT (PINC.2 == 0)
#define BUTTON_RIGHT (PINC.3 == 0)
#define BUTTON_SELECT (PINC.4 == 0)
#define BUTTON_ONOFF (PINC.5 == 0)

#define BUTTON_LED PORTC.6 //Notifikační LED, log.1 svítí

#define BUTTON_START PINE.7
#define BUTTON_STOP PINE.6

//Motor - lin. posuv TB6560 spínáno přes optočleny, přepnutím do nuly --> optočlen sepne
#define STEP_MOVE PORTB.4 //defaultně 0
#define DIRECTION_MOVE PORTB.5 //defaultně 0
#define ENABLE_MOVE PORTB.6 //defaultně 0, driver je vypnutý

//Motor - rot. pohyb DRV8825
#define STEP_ROT PORTD.5 //defaultně 0
#define DIRECTION_ROT PORTD.4 //defaultně 0
#define ENABLE_ROT PORTE.3 //defaultně 1, driver je VYPNUTÝ

//Poloha vozíku - zpětná vazba
#define PHASE_LIN_A PIND.1 //interrupt INT1
#define PHASE_LIN_B PIND.7 //čítač T2

//Rotace kamery - zpětná vazba
#define PHASE_ROT_A PIND.0 //interrupt INT1
#define PHASE_ROT_B PIND.6 //čítač T2

```

```

//Bluetooth
#define BT_ENB                PORTE.2        //Zapnutí modulu (defaultně 1)
//-----

// External Interrupt 4, VOLNOBĚŽNÁ KLADKA
interrupt [EXT_INT4] void ext_int4_isr(void)
{
    if(DIRECTION_MOVE == 0)
    {
        step_mov = 0;
        BUTTON_LED = 1;
        DIRECTION_MOVE = 1;
        item = 2;
        ETIMSK=0x00;
        TCCR0=0x00;
    }
}

// External Interrupt 5, STRANA U MOTORU
interrupt [EXT_INT5] void ext_int5_isr(void)
{
    if(DIRECTION_MOVE)
    {
        step_mov = 0;
        BUTTON_LED = 1;
        DIRECTION_MOVE = 0;
        item = 1;
        ETIMSK=0x00;
        TCCR0=0x00;
    }
}

// External Interrupt 6 service routine -- STOP TLAČITKO
interrupt [EXT_INT6] void ext_int6_isr(void)
{
    TCCR0=0x05;
    ETIMSK= 0x00;
    uvod = 1;
    switch(menu)
    {
        case FOTO:
            break;
        case VIDEO:
            break;
        case BLUETOOTH:
            break;
        case SNIMKOVANI:
            menu = FOTO;
            break;
        case TIMELAPSE:
            menu = FOTO;
            break;
        case NATACENI:
            menu = VIDEO;
            break;
    }
}

// External Interrupt 7 service routine -- START TLAČITKO
interrupt [EXT_INT7] void ext_int7_isr(void)

```

```

{
  TCCR0=0x05;
  ETIMSK= 0x04;
  uvod = 1;
  prerus = 1;
  ENABLE_MOVE = 1;
  ENABLE_ROT = 0;
  switch(menu)
  {
    case FOTO:
      menu = TIMELAPSE;
      break;
    case VIDEO:
      menu = NATACENI;
      break;
    case BLUETOOTH:
      break;
    case SNIMKOVANI:
      menu = TIMELAPSE;
      break;
    case TIMELAPSE:
      break;
    case NATACENI:
      break;
  }
}

// Timer 0 overflow interrupt service routine
interrupt [TIM0_OVF] void timer0_ovf_isr(void)
{
  // Place your code here

  time_ms++;

  if((time_ms % 4) == 0) //250ms
  {
    time++; //1 sekunda
  }
  if(time > 359999) //100*60*60-1 maximální zobrazitelná hodnota
  {
    time = 0;
    M = 0;
    H = 0;
  }

  if(time_ms > 359999)
  {
    time_ms = 0;
  }
}

// Timer3 overflow interrupt service routine
interrupt [TIM3_OVF] void timer3_ovf_isr(void)
{
  // Reinitialize Timer3 value
  //TCNT3H=0xE700 >> 8;
  //TCNT3L=0xE700 & 0xff;
  TCNT3H=0xF380 >> 8;
  TCNT3L=0xF380 & 0xff;
  // Place your code here
}

```



```

step_mov++;
step_rot++;
fast_step_move = !(fast_step_move);
if ((step_mov > def_step_mov) && (prerus))
{
    STEP_MOVE = 1;
    BUTTON_LED = 1;
    step_mov = 0;
    per_step++;
}

else
{
    STEP_MOVE = 0;
    BUTTON_LED = 0;
}

if (step_rot > def_step_rot)
{
    STEP_ROT = 1;
    BUTTON_LED = 1;
    step_rot = 0;
}

else
{
    STEP_ROT = 0;
    BUTTON_LED = 0;
}

if(menu == TIMELAPSE)
{
    if(per_step >= step_of_shot)
    {
        ETIMSK= 0x00;
        PORTF &= ~(1<<2);    //ostření = 0, zaostřeno
        delay_ms(80);
        PORTF &= ~(1<<3);    //spoušť = 0, vyfocení
        delay_ms(800);      //ČAS PRO VLOŽENÍ DÉLKY EXPOZICE
        PORTF |= (1<<3);    //spoušť = 1, vypnuto
        PORTF |= (1<<2);    //ostření = 1, vypnuto
        per_step = 0;
        ETIMSK= 0x04;
    }
}

if(BUTTON_RIGHT)
{
    DIRECTION_MOVE = 0;
    item = 1;
    ENABLE_MOVE = 1;
    prerus = 0;
    STEP_MOVE = fast_step_move;    //maximální frekvence t-on je poloviční (1500Hz)
    /*
    while(fast_step_move == 1)
    {
        STEP_MOVE = fast_step_move;
        if((BUTTON_STOP != 1) || (LIMIT_SWITCH_START != 1))
            break;
    }
}

```

```

    */
    //prerus = 1;
    //while(BUTTON_RIGHT);
}

if(BUTTON_LEFT)
{
    DIRECTION_MOVE = 1;
    item = 2;
    ENABLE_MOVE = 1;
    prerus = 0;
    STEP_MOVE = fast_step_move;
}

}

void main(void)
{
// Declare your local variables here
    GLCDINIT_t init;

// Variable used to store graphic display
// controller initialization data
    GLCDINIT_t glcd_init_data;

// Input/Output Ports initialization
// Port A initialization
// Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
    DDRA=(0<<DDA7) | (0<<DDA6) | (0<<DDA5) | (0<<DDA4) | (0<<DDA3) | (0<<DDA2) | (0<<DDA1) |
    (0<<DDA0);
// State: Bit7=T Bit6=T Bit5=T Bit4=T Bit3=T Bit2=T Bit1=T Bit0=T
    PORTA=(0<<PORTA7) | (0<<PORTA6) | (0<<PORTA5) | (0<<PORTA4) | (0<<PORTA3) | (0<<PORTA2) |
    (0<<PORTA1) | (0<<PORTA0);

// Port B initialization
// Function: Bit7=In Bit6=Out Bit5=Out Bit4=Out Bit3=In Bit2=In Bit1=In Bit0=In
    DDRB=(0<<DDB7) | (1<<DDB6) | (1<<DDB5) | (1<<DDB4) | (0<<DDB3) | (0<<DDB2) | (0<<DDB1) |
    (0<<DDB0);
// State: Bit7=T Bit6=0 Bit5=1 Bit4=1 Bit3=T Bit2=T Bit1=T Bit0=T
    PORTB=(0<<PORTB7) | (0<<PORTB6) | (1<<PORTB5) | (1<<PORTB4) | (0<<PORTB3) | (0<<PORTB2) |
    (0<<PORTB1) | (0<<PORTB0);

// Port C initialization
// Function: Bit7=In Bit6=Out Bit5=In Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
    DDRC=(0<<DDC7) | (1<<DDC6) | (0<<DDC5) | (0<<DDC4) | (0<<DDC3) | (0<<DDC2) | (0<<DDC1) |
    (0<<DDC0);
// State: Bit7=P Bit6=1 Bit5=P Bit4=P Bit3=P Bit2=P Bit1=P Bit0=P
    PORTC=(1<<PORTC7) | (1<<PORTC6) | (1<<PORTC5) | (1<<PORTC4) | (1<<PORTC3) | (1<<PORTC2) |
    (1<<PORTC1) | (1<<PORTC0);

// Port D initialization
// Function: Bit7=In Bit6=In Bit5=Out Bit4=Out Bit3=In Bit2=In Bit1=In Bit0=In
    DDRD=(0<<DDD7) | (0<<DDD6) | (1<<DDD5) | (1<<DDD4) | (0<<DDD3) | (0<<DDD2) | (0<<DDD1) |
    (0<<DDD0);
// State: Bit7=T Bit6=T Bit5=0 Bit4=0 Bit3=P Bit2=P Bit1=T Bit0=T
    PORTD=(0<<PORTD7) | (0<<PORTD6) | (0<<PORTD5) | (0<<PORTD4) | (1<<PORTD3) | (1<<PORTD2) |
    (0<<PORTD1) | (0<<PORTD0);

// Port E initialization
// Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=Out Bit2=Out Bit1=In Bit0=In
    DDRE=(0<<DDE7) | (0<<DDE6) | (0<<DDE5) | (0<<DDE4) | (1<<DDE3) | (1<<DDE2) | (0<<DDE1) |

```

```

(0<<DDE0);
// State: Bit7=P Bit6=P Bit5=P Bit4=P Bit3=1 Bit2=1 Bit1=T Bit0=T
PORTE=(1<<PORTE7) | (1<<PORTE6) | (1<<PORTE5) | (1<<PORTE4) | (1<<PORTE3) | (1<<PORTE2) |
(0<<PORTE1) | (0<<PORTE0);

// Port F initialization
// Function: Bit7=In Bit6=In Bit5=In Bit4=In Bit3=Out Bit2=Out Bit1=Out Bit0=Out
DDRF=(0<<DDF7) | (0<<DDF6) | (0<<DDF5) | (0<<DDF4) | (1<<DDF3) | (1<<DDF2) | (1<<DDF1) |
(1<<DDF0);
// State: Bit7=P Bit6=P Bit5=P Bit4=P Bit3=1 Bit2=1 Bit1=1 Bit0=1
PORTF=(1<<PORTF7) | (1<<PORTF6) | (1<<PORTF5) | (1<<PORTF4) | (1<<PORTF3) | (1<<PORTF2) |
(1<<PORTF1) | (1<<PORTF0);

// Port G initialization
// Function: Bit4=In Bit3=In Bit2=In Bit1=In Bit0=In
DDRG=(0<<DDG4) | (0<<DDG3) | (0<<DDG2) | (0<<DDG1) | (0<<DDG0);
// State: Bit4=T Bit3=T Bit2=P Bit1=T Bit0=T
PORTG=(0<<PORTG4) | (0<<PORTG3) | (1<<PORTG2) | (0<<PORTG1) | (0<<PORTG0);

// Timer/Counter 0 initialization
// Clock source: TOSC1 pin
// Clock value: TOSC1/32
// Mode: Normal top=0xFF
// OC0 output: Disconnected
ASSR=1<<AS0;
TCCR0=(0<<WGM00) | (0<<COM01) | (0<<COM00) | (0<<WGM01) | (0<<CS02) | (1<<CS01) | (1<<CS00);
TCNT0=0x00;
OCR0=0x40;

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer1 Stopped
// Mode: Normal top=0xFFFF
// OC1A output: Disconnected
// OC1B output: Disconnected
// OC1C output: Disconnected
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
// Compare C Match Interrupt: Off
TCCR1A=(0<<COM1A1) | (0<<COM1A0) | (0<<COM1B1) | (0<<COM1B0) | (0<<COM1C1) |
(0<<COM1C0) | (0<<WGM11) | (0<<WGM10);
TCCR1B=(0<<ICNC1) | (0<<ICES1) | (0<<WGM13) | (0<<WGM12) | (0<<CS12) | (0<<CS11) | (0<<CS10);
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;
OCR1CH=0x00;
OCR1CL=0x00;

// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer2 Stopped

```

```
// Mode: Normal top=0xFF
// OC2 output: Disconnected
TCCR2=(0<<WGM20) | (0<<COM21) | (0<<COM20) | (0<<WGM21) | (0<<CS22) | (0<<CS21) | (0<<CS20);
TCNT2=0x00;
OCR2=0x00;

// Timer/Counter 3 initialization
// Clock source: System Clock
// Clock value: 16000,000 kHz
// Mode: Normal top=0xFFFF
// OC3A output: Disconnected
// OC3B output: Disconnected
// OC3C output: Disconnected
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer Period: 0,4 ms
// Timer3 Overflow Interrupt: On
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
// Compare C Match Interrupt: Off
TCCR3A=(0<<COM3A1) | (0<<COM3A0) | (0<<COM3B1) | (0<<COM3B0) | (0<<COM3C1) |
(0<<COM3C0) | (0<<WGM31) | (0<<WGM30);
TCCR3B=(0<<ICNC3) | (0<<ICES3) | (0<<WGM33) | (0<<WGM32) | (0<<CS32) | (0<<CS31) | (1<<CS30);
TCNT3H=0xE7;
TCNT3L=0x00;
ICR3H=0x00;
ICR3L=0x00;
OCR3AH=0x00;
OCR3AL=0x00;
OCR3BH=0x00;
OCR3BL=0x00;
OCR3CH=0x00;
OCR3CL=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=(0<<OCIE2) | (0<<TOIE2) | (0<<TICIE1) | (0<<OCIE1A) | (0<<OCIE1B) | (0<<TOIE1) | (0<<OCIE0)
| (1<<TOIE0);
ETIMSK=(0<<TICIE3) | (0<<OCIE3A) | (0<<OCIE3B) | (0<<TOIE3) | (0<<OCIE3C) | (0<<OCIE1C);

// External Interrupt(s) initialization
// INT0: On
// INT0 Mode: Rising Edge
// INT1: On
// INT1 Mode: Rising Edge
// INT2: Off
// INT3: Off
// INT4: On
// INT4 Mode: Falling Edge
// INT5: On
// INT5 Mode: Falling Edge
// INT6: On
// INT6 Mode: Low level
// INT7: On
// INT7 Mode: Low level
EICRA=(0<<ISC31) | (0<<ISC30) | (0<<ISC21) | (0<<ISC20) | (1<<ISC11) | (1<<ISC10) | (1<<ISC01) |
(1<<ISC00);
EICRB=(0<<ISC71) | (0<<ISC70) | (0<<ISC61) | (0<<ISC60) | (1<<ISC51) | (0<<ISC50) | (1<<ISC41) |
(0<<ISC40);
```

```

EIMSK=(1<<INT7) | (1<<INT6) | (1<<INT5) | (1<<INT4) | (0<<INT3) | (0<<INT2) | (1<<INT1) | (1<<INT0);
EIFR=(1<<INTF7) | (1<<INTF6) | (1<<INTF5) | (1<<INTF4) | (0<<INTF3) | (0<<INTF2) | (1<<INTF1) |
(1<<INTF0);

// USART0 initialization
// USART0 disabled
UCSR0B=(0<<RXCIE0) | (0<<TXCIE0) | (0<<UDRIE0) | (0<<RXEN0) | (0<<TXEN0) | (0<<UCSZ02) |
(0<<RXB80) | (0<<TXB80);

// USART1 initialization
// USART1 disabled
UCSR1B=(0<<RXCIE1) | (0<<TXCIE1) | (0<<UDRIE1) | (0<<RXEN1) | (0<<TXEN1) | (0<<UCSZ12) |
(0<<RXB81) | (0<<TXB81);

// Analog Comparator initialization
// Analog Comparator: Off
// The Analog Comparator's positive input is
// connected to the AIN0 pin
// The Analog Comparator's negative input is
// connected to the AIN1 pin
ACSR=(1<<ACD) | (0<<ACBG) | (0<<ACO) | (0<<ACI) | (0<<ACIE) | (0<<ACIC) | (0<<ACIS1) |
(0<<ACIS0);
SFIOR=(0<<ACME);

// ADC initialization
// ADC disabled
ADCSRA=(0<<ADEN) | (0<<ADSC) | (0<<ADFR) | (0<<ADIF) | (0<<ADIE) | (0<<ADPS2) | (0<<ADPS1) |
(0<<ADPS0);

// SPI initialization
// SPI disabled
SPCR=(0<<SPIE) | (0<<SPE) | (0<<DORD) | (0<<MSTR) | (0<<CPOL) | (0<<CPHA) | (0<<SPR1) |
(0<<SPR0);

// TWI initialization
// TWI disabled
TWCR=(0<<TWEA) | (0<<TWSTA) | (0<<TWSTO) | (0<<TWEN) | (0<<TWIE);

// Graphic Display Controller initialization
// The ST7920 connections are specified in the
// Project|Configure|C Compiler|Libraries|Graphic Display menu:
// E - PORTA Bit 5
// R /W - PORTA Bit 6
// RS - PORTA Bit 7
// /RST - PORTA Bit 0
// DB4 - PORTA Bit 4
// DB5 - PORTA Bit 3
// DB6 - PORTA Bit 2
// DB7 - PORTA Bit 1

// Specify the current font for displaying text
glcd_init_data.font=font5x7;
// No function is used for reading
// image data from external memory
glcd_init_data.readxmem=NULL;
// No function is used for writing
// image data to external memory
glcd_init_data.writexmem=NULL;

glcd_init(&glcd_init_data);

```

```

//-----
/* Set current font */
init.font=font5x7;
/* No need for reading data from external memory */
init.readxmem=NULL;

/* No need for reading data from external memory */
init.writexmem=NULL;

/* Initialize the LCD controller and graphics */
glcd_init(&init);
//-----

// Global enable interrupts
#asm("sei")

while (1)
{
    // Place your code here
    S = time % 60;
    last_delta_time = time;

    if(time != last_time)
    {
        last_time = time;
        if((time % 60) == 0)
        {
            M++;
        }
        if((time % 3600) == 0)
        {
            H++;
            M = 0;
        }
    }
}

if (BUTTON_MENU)
{
    menu++;
    uvod = 1;
    if(menu > 2)
    {
        menu = 0;
        uvod = 1;
    }
    while (BUTTON_MENU);
}

switch(menu)
{
    case FOTO:
        if(uvod)
        {
            glcd_clear();
            uvod = 0;
            position = 12;
        }
        glcd_line(0,0,125,0);
        glcd_moveto(0,1);
}

```

```
glcd_setcolor(0);
glcd_setbkcolor(1);
glcd_setfont(font5x7);
glcd_outtextf("Foto rezim 1. prehled");
//glcd_line(0,9,125,9);

glcd_moveto(0,12);
glcd_setcolor(1);
glcd_setbkcolor(0);
glcd_settextjustify(1,2);
glcd_outtextf("Draha  \n");
if(item == 1)
{
    glcd_outtextxyf(75,12,"mot->bat \n");
    DIRECTION_MOVE = 0;
}
else if (item == 2)
{
    glcd_outtextxyf(75,12,"bat->mot \n");
    DIRECTION_MOVE = 1;
}
else
    glcd_outtextxyf(75,12," vlastni \n");
//glcd_outtextxyf(75,12,"mot->bat \n");
//glcd_outtextxyf(75,12,"bat->mot \n");
//glcd_outtextxyf(75,12," vlastni \n");

glcd_moveto(0,21);
glcd_outtextf("Snimkovani  \n");
glcd_outtextxyf(84,21," 256\n");

glcd_outtextf("Posuv      Hz\n");
glcd_outtextxyf(85,30," 100\n");

glcd_outtextf("Rotace      Hz\n");
glcd_outtextxyf(85,39," 100\n");

glcd_outtextf("Otaceni      \n");
if(DIRECTION_ROT == 0)
    glcd_outtextxyf(80,48," doleva\n");
else
    glcd_outtextxyf(80,48," doprava\n");

glcd_outtextf("Odlozeni startu 0:00");
//glcd_outtextxyf(100,57,"0:00");

glcd_setcolor(0);
glcd_line(127,10,127,63);
glcd_line(126,10,126,63);
glcd_setcolor(1);

break;

case VIDEO:
    if(uvod)
    {
```

```

        glcd_clear();
        uvod = 0;
        position = 12;
    }
    glcd_moveto(0,1);
    glcd_setcolor(0);
    glcd_setbkcolor(1);
    glcd_setfont(font5x7);
    glcd_outtextf("Videorezim - manual \n");
    //glcd_line(0,9,125,9);

    glcd_moveto(0,12);
    glcd_setcolor(1);
    glcd_setbkcolor(0);
    glcd_settextjustify(1,2);
    glcd_outtextf("Draha\n");    //motor, kladka, vlastný
    if(item == 1)
    {
        glcd_outtextxyf(75,12,"mot->bat \n");
        DIRECTION_MOVE = 0;
    }
    else if (item == 2)
    {
        glcd_outtextxyf(75,12,"bat->mot \n");
        DIRECTION_MOVE = 1;
    }
    else
        glcd_outtextxyf(75,12," vlastni \n");

    glcd_moveto(0,21);
    glcd_outtextf("Posuv      Hz\n");
    glcd_outtextxyf(85,21,"1000\n");

    glcd_outtextf("Rotace      Hz\n");
    glcd_outtextxyf(85,30," 100\n");

    glcd_outtextf("Otaceni      \n");
    if(DIRECTION_ROT == 0)
        glcd_outtextxyf(80,39," doleva\n");
    else
        glcd_outtextxyf(80,39," doprava\n");
    glcd_setcolor(0);
    glcd_line(127,10,127,63);
    glcd_line(126,10,126,63);
    glcd_setcolor(1);

break;

case BLUETOOTH:
    if(uvod)
    {
        glcd_clear();
        uvod = 0;
        position = 12;
    }

    glcd_line(0,0,125,0);
    //glcd_line(0,0,125,0);

```



```

glcd_moveto(0,1);
glcd_setcolor(0);
glcd_setbkcolor(1);
glcd_setfont(font5x7);
glcd_outtextf("Bluetooth - ovladac \n");
//glcd_line(0,9,125,9);

glcd_moveto(0,12);
glcd_setcolor(1);
glcd_setbkcolor(0);
glcd_settextjustify(1,2);
glcd_outtextf("Posuv      Hz\n");
glcd_outtextxyf(85,12,"2500\n");

glcd_outtextf("Rotace      Hz\n");
glcd_outtextxyf(85,21,"100\n");

glcd_setcolor(0);
glcd_line(127,10,127,63);
glcd_line(126,10,126,63);
glcd_setcolor(1);

break;

case SNIMKOVANI:
  if(uvod)
  {
    glcd_clear();
    uvod = 0;
    position = 12;
  }

  glcd_line(0,0,125,0);
  glcd_moveto(0,1);
  glcd_setcolor(0);
  glcd_setbkcolor(1);
  glcd_setfont(font5x7);
  glcd_outtextf("Snimkovani:  \n");
  //glcd_line(0,9,125,9);

  glcd_moveto(0,12);
  glcd_setcolor(1);
  glcd_setbkcolor(0);
  glcd_settextjustify(1,2);
  glcd_outtextf("Snimky\n");
  glcd_outtextxyf(80,12,"256\n");

  glcd_moveto(0,21);
  glcd_outtextf("Expozice      s\n");
  glcd_outtextxyf(100,21,"3,3\n");

  glcd_outtextf("Cas videa      s\n");
  glcd_outtextxyf(100,30,"8,1\n");

  glcd_outtextf("Perioda sn.    s\n");
  glcd_outtextxyf(100,39,"3,3\n");

```

```
glcd_outtextf("Celkovy cas      \n");
glcd_outtextf("                \n");
glcd_outtextxyf(50,57,"      \n");
glcd_outtextxyf(80,57,"    min\n");
//glcd_outtextxyf(80,48,"doprava\n");

glcd_setcolor(0);
glcd_line(127,10,127,45);
glcd_line(126,10,126,45);
glcd_setcolor(1);

break;

case TIMELAPSE:
if(uvod)
{
    glcd_clear();
    uvod = 0;
    time = 0;
    S = 0;
    M = 0;
    H = 0;
    position = 12;
}

glcd_line(0,0,125,0);
glcd_moveto(0,1);
glcd_setcolor(0);
glcd_setbkcolor(1);
glcd_setfont(font5x7);
glcd_outtextf("Timelapse - ");
numToStr(H, string);
glcd_outtext(string);
glcd_outtextf(":");
numToStr(M, string);
glcd_outtext(string);
glcd_outtextf(":");
numToStr(S, string);
glcd_outtext(string);
//glcd_line(0,9,125,9);

glcd_moveto(0,12);
glcd_setcolor(1);
glcd_setbkcolor(0);
glcd_settextjustify(1,2);
glcd_outtextf("Snimky\n");
glcd_outtextxyf(80,12," 256\n");

glcd_moveto(0,21);
glcd_outtextf("Expozice      s\n");
glcd_outtextxyf(100,21,"3,3\n");

glcd_outtextf("Perioda sn.      s\n");
glcd_outtextxyf(100,30,"8,3\n");

glcd_outtextf("Posuv          Hz\n");
glcd_outtextxyf(85,39,"1000\n");
```

```

glcd_outtextf("Rotace      Hz\n");
glcd_outtextxyf(85,48," 100\n");

glcd_outtextf("Otaceni      \n");
if(DIRECTION_ROT == 0)
    glcd_outtextxyf(80,57," doleva\n");
else
    glcd_outtextxyf(80,57," doprava\n");

glcd_setcolor(0);
glcd_line(127,10,127,63);
glcd_line(126,10,126,63);
glcd_setcolor(1);

break;

case NATACENI:
if(uvod)
{
    glcd_clear();
    uvod = 0;
    time = 0;
    S = 0;
    M = 0;
    H = 0;
    position = 12;
}

glcd_line(0,0,125,0);
glcd_moveto(0,1);
glcd_setcolor(0);
glcd_setbkcolor(1);
glcd_setfont(font5x7);
glcd_outtextf("Nataceni - ");
numToStr(H, string);
glcd_outtext(string);
glcd_outtextf(":");
numToStr(M, string);
glcd_outtext(string);
glcd_outtextf(":");
numToStr(S, string);
glcd_outtext(string);
//glcd_line(0,9,125,9);

glcd_moveto(0,12);
glcd_setcolor(1);
glcd_setbkcolor(0);
glcd_settextjustify(1,2);
glcd_outtextf("Smer      \n");
glcd_outtextxyf(80,12," <|\n");
if(DIRECTION_MOVE == 0)
    glcd_outtextxyf(70,12,"< doleva \n");
else
    glcd_outtextxyf(70,12,"< doprava\n");
//glcd_outtextxyf(80,12," <|\n");

```

```

glcd_moveto(0,21);
glcd_outtextf("Otaceni      \n");
if(DIRECTION_ROT == 0)
    glcd_outtextxyf(70,21,"< doleva \n");
else
    glcd_outtextxyf(70,21,"< doprava\n");
//glcd_outtextxyf(80,21," <|>\n");

glcd_outtextf("Posuv      Hz\n");
glcd_outtextxyf(85,30,"1000\n");

glcd_outtextf("Rotace      Hz\n");
glcd_outtextxyf(85,39," 100\n");

//glcd_outtextf("      \n");
//glcd_outtextf("      \n");

glcd_setcolor(0);
glcd_line(127,10,127,63);
glcd_line(126,10,126,63);
glcd_setcolor(1);

break;
}

switch(menu)
{
case FOTO:
    if(BUTTON_UP)
    {
        position -= 9;
        if(position < 12)
        {
            position = 57;
        }

        while(BUTTON_UP);
    }

    if(BUTTON_DOWN)
    {
        position += 9;
        if(position > 57)
        {
            position = 12;
        }

        while(BUTTON_DOWN);
    }
    /* Line thickness: 7 pixels */
    glcd_setlinestyle(7,GLCD_LINE_SOLID);
    glcd_line(126,position,127,position);

    if(!(PING & (1<<2)))
    {
        menu=3;
        uvod=1;
    }
}

```

```
}
while(!(PING & (1<<2)));

if((position == 12) && (BUTTON_NEXT))
{
    item++;
    if(item>3)
        item=1;
    while(BUTTON_NEXT);
}

if((position == 12) && (BUTTON_BEFORE))
{
    item--;
    if(item<1)
        item=3;
    while(BUTTON_BEFORE);
}

if((position == 30) && (BUTTON_NEXT))
{
    if((def_step_mov <= 10) && (def_step_mov >= 1))
        def_step_mov -= 1;
    else if((def_step_mov <= 199) && (def_step_mov >= 11))
        def_step_mov -= 25;
    else if((def_step_mov <= 499) && (def_step_mov >= 200))
        def_step_mov -= 100;
    else if((def_step_mov <= 1500) && (def_step_mov >= 500))
        def_step_mov -= 500;

    if(def_step_mov < 1)
        def_step_mov = 5;
    if(def_step_mov > 1500)
        def_step_mov = 1500;
    //while(BUTTON_NEXT);
}

if((position == 30) && (BUTTON_BEFORE))
{
    if((def_step_mov <= 10) && (def_step_mov >= 1))
        def_step_mov += 1;
    else if((def_step_mov <= 199) && (def_step_mov >= 11))
        def_step_mov += 25;
    else if((def_step_mov <= 499) && (def_step_mov >= 200))
        def_step_mov += 100;
    else if((def_step_mov <= 1500) && (def_step_mov >= 500))
        def_step_mov += 500;

    if(def_step_mov > 1500)
        def_step_mov = 1500;
    if(def_step_mov < 1)
        def_step_mov = 1;
    //while(BUTTON_BEFORE);
}

if((position == 39) && (BUTTON_NEXT))
{
    if((def_step_mov <= 1000) && (def_step_mov >= 700))
        def_step_mov -= 100;
    else if((def_step_mov <= 3000) && (def_step_mov >= 1001))
```

```
    def_step_mov -= 250;

    if(def_step_mov > 3000)
        def_step_mov = 3000;
    if(def_step_mov < 700)
        def_step_mov = 700;
    //while(BUTTON_NEXT);
}

if((position == 39) && (BUTTON_BEFORE))
{
    if((def_step_mov <= 1000) && (def_step_mov >= 700))
        def_step_mov += 100;
    else if((def_step_mov <= 3000) && (def_step_mov >= 1001))
        def_step_mov += 250;

    if(def_step_mov > 3000)
        def_step_mov = 3000;
    if(def_step_mov < 700)
        def_step_mov = 700;
    //while(BUTTON_BEFORE);
}

if((position == 48) && (BUTTON_NEXT))
{
    DIRECTION_ROT = 1;
    while(BUTTON_NEXT);
}

if((position == 48) && (BUTTON_BEFORE))
{
    DIRECTION_ROT = 0;
    while(BUTTON_BEFORE);
}

break;

case VIDEO:
    if(BUTTON_UP)
    {
        position -= 9;
        if(position < 12)
        {
            position = 39;
        }

        while(BUTTON_UP);
    }

    if(BUTTON_DOWN)
    {
        position += 9;
        if(position > 39)
        {
            position = 12;
        }

        while(BUTTON_DOWN);
    }
}
```

```
}
/* Line thickness: 7 pixels */
glcd_setlinestyle(7,GLCD_LINE_SOLID);
glcd_line(126,position,127,position);

if((position == 12) && (BUTTON_NEXT))
{
    item++;
    if(item>3)
        item=1;
    while(BUTTON_NEXT);
}

if((position == 12) && (BUTTON_BEFORE))
{
    item--;
    if(item<1)
        item=3;
    while(BUTTON_BEFORE);
}

if((position == 21) && (BUTTON_NEXT))
{
    def_step_mov -= 50;
    if(def_step_mov < 1)
        def_step_mov = 500;
    //while(BUTTON_NEXT);
}

if((position == 21) && (BUTTON_BEFORE))
{
    def_step_mov += 50;
    if(def_step_mov > 500)
        def_step_mov = 1;
    //while(BUTTON_BEFORE);
}

if((position == 30) && (BUTTON_NEXT))
{
    def_step_rot -= 50;
    if(def_step_rot < 1)
        def_step_rot = 800;
    //while(BUTTON_NEXT);
}

if((position == 30) && (BUTTON_BEFORE))
{
    def_step_rot += 50;
    if(def_step_rot > 800)
        def_step_rot = 1;
    //while(BUTTON_BEFORE);
}

if((position == 39) && (BUTTON_NEXT))
{
    DIRECTION_ROT = 1;
    while(BUTTON_NEXT);
}

if((position == 39) && (BUTTON_BEFORE))
```

```

    {
        DIRECTION_ROT = 0;
        while(BUTTON_BEFORE);
    }
break;

case BLUETOOTH:
if(BUTTON_UP)
{
    position -= 9;
    if(position < 12)
    {
        position = 21;
    }
    while(BUTTON_UP);
}

if(BUTTON_DOWN)
{
    position += 9;
    if(position > 21)
    {
        position = 12;
    }
    while(BUTTON_DOWN);
}
/* Line thickness: 7 pixels */
glcd_setlinestyle(7, GLCD_LINE_SOLID);
glcd_line(126, position, 127, position);

if((position == 12) && (BUTTON_NEXT))
{
    def_step_mov -= 50;
    if(def_step_mov < 1)
        def_step_mov = 500;
    //while(BUTTON_NEXT);
}

if((position == 12) && (BUTTON_BEFORE))
{
    def_step_mov += 50;
    if(def_step_mov > 500)
        def_step_mov = 1;
    //while(BUTTON_BEFORE);
}

if((position == 21) && (BUTTON_NEXT))
{
    def_step_rot -= 50;
    if(def_step_rot < 1)
        def_step_rot = 800;
    //while(BUTTON_NEXT);
}

if((position == 21) && (BUTTON_BEFORE))
{
    def_step_rot += 50;
    if(def_step_rot > 800)
        def_step_rot = 1;
}

```



```

    //while(BUTTON_BEFORE);
  }
break;

case SNIMKOVANI:
  if(BUTTON_UP)
  {
    position -= 9;
    if(position < 12)
    {
      position = 21;
    }

    while(BUTTON_UP);
  }

  if(BUTTON_DOWN)
  {

    position += 9;
    if(position > 21)
    {
      position = 12;
    }

    while(BUTTON_DOWN);
  }
  /* Line thickness: 7 pixels */
  glcd_setlinestyle(7,GLCD_LINE_SOLID);
  glcd_line(126,position,127,position);

  if(!(PING & (1<<2))) //enter
  {
    menu=FOTO;
    uvod=1;
  }
  while(!(PING & (1<<2)));

  if((position == 12) && (BUTTON_NEXT))
  {
    picture+=12;
    while(BUTTON_NEXT);
  }

  if((position == 12) && (BUTTON_BEFORE))
  {
    picture-=12;
    while(BUTTON_BEFORE);
  }

  if((position == 21) && (BUTTON_NEXT))
  {
    expozice+=1; //incrementace po desetínách ???
    while(BUTTON_NEXT);
  }

  if((position == 21) && (BUTTON_BEFORE))
  {
    expozice-=1;
    while(BUTTON_BEFORE);
  }

```

```

    }

    time_move = (2560000/(pull*def_step_mov)); //pomocný výpočet
    per_of_shot = time_move/picture; // zobrazení na displeji
    //step_of_shot = 1/per_of_shot; //přerušení čítače, udělení snímku dle délky expozice,
    // pokračování ve snímání
    total_time = time_move + (picture * expozice); // zobrazení na displeji

break;

case TIMELAPSE:
if(BUTTON_UP)
{
    position -= 9;
    if(position < 12)
    {
        position = 57;
    }

    while(BUTTON_UP);
}

if(BUTTON_DOWN)
{
    position += 9;
    if(position > 57)
    {
        position = 12;
    }

    while(BUTTON_DOWN);
}
/* Line thickness: 7 pixels */
glcd_setlinestyle(7,GLCD_LINE_SOLID);
glcd_line(126,position,127,position);

if((position == 12) && (BUTTON_NEXT))
{
    picture+=12;
    while(BUTTON_NEXT);
}

if((position == 12) && (BUTTON_BEFORE))
{
    picture-=12;
    while(BUTTON_BEFORE);
}

if((position == 21) && (BUTTON_NEXT))
{
    expozice+=1; //incrementace po desetínách ???
    while(BUTTON_NEXT);
}

if((position == 21) && (BUTTON_BEFORE))
{
    expozice-=1;
    while(BUTTON_BEFORE);
}

```

```
if(!(PING & (1<<2))) //enter
{
    time_move = (2560000/(pull*def_step_mov)); //pomocný výpočet
    per_of_shot = time_move/picture; // zobrazení na displeji
    //step_of_shot = 1/per_of_shot; //přerušení čítače, udělení snímku dle délky expozice,
    //pokračování ve snímání
    total_time = time_move + (picture * expozice); // zobrazení na displeji
}

if((position == 39) && (BUTTON_NEXT))
{
    def_step_mov -= 50;
    if(def_step_mov < 1)
        def_step_mov = 500;
    //while(BUTTON_NEXT);
}

if((position == 39) && (BUTTON_BEFORE))
{
    def_step_mov += 50;
    if(def_step_mov > 500)
        def_step_mov = 1;
    //while(BUTTON_BEFORE);
}

if((position == 48) && (BUTTON_NEXT))
{
    def_step_rot -= 50;
    if(def_step_rot < 1)
        def_step_rot = 800;
    //while(BUTTON_NEXT);
}

if((position == 48) && (BUTTON_BEFORE))
{
    def_step_rot += 50;
    if(def_step_rot > 800)
        def_step_rot = 1;
    //while(BUTTON_BEFORE);
}

if((position == 57) && (BUTTON_NEXT))
{
    DIRECTION_ROT = 1;
    while(BUTTON_NEXT);
}

if((position == 57) && (BUTTON_BEFORE))
{
    DIRECTION_ROT = 0;
    while(BUTTON_BEFORE);
}

break;

case NATACENI:
    if(BUTTON_UP)
    {
```

```
    position -= 9;
    if(position < 12)
    {
        position = 39;
    }

    while(BUTTON_UP);
}

if(BUTTON_DOWN)
{

    position += 9;
    if(position > 39)
    {
        position = 12;
    }

    while(BUTTON_DOWN);
}
/* Line thickness: 7 pixels */
glcd_setlinestyle(7, GLCD_LINE_SOLID);
glcd_line(126, position, 127, position);

if((position == 12) && (BUTTON_NEXT))
{
    DIRECTION_MOVE = 1;
    while(BUTTON_NEXT);
}

if((position == 12) && (BUTTON_BEFORE))
{
    DIRECTION_MOVE = 0;
    while(BUTTON_BEFORE);
}

if((position == 21) && (BUTTON_NEXT))
{
    DIRECTION_ROT = 1;
    while(BUTTON_NEXT);
}

if((position == 21) && (BUTTON_BEFORE))
{
    DIRECTION_ROT = 0;
    while(BUTTON_BEFORE);
}

if((position == 30) && (BUTTON_NEXT))
{
    def_step_mov -= 50;
    if(def_step_mov < 1)
        def_step_mov = 500;
    //while(BUTTON_NEXT);
}

if((position == 30) && (BUTTON_BEFORE))
{
    def_step_mov += 50;
    if(def_step_mov > 500)
```

```

        def_step_mov = 1;
        //while(BUTTON_BEFORE);
    }

    if((position == 39) && (BUTTON_NEXT))
    {
        def_step_rot -= 50;
        if(def_step_rot < 1)
            def_step_rot = 800;
        //while(BUTTON_NEXT);
    }

    if((position == 39) && (BUTTON_BEFORE))
    {
        def_step_rot += 50;
        if(def_step_rot > 800)
            def_step_rot = 1;
        //while(BUTTON_BEFORE);
    }
    break;
}

//BUTTON_LED = 0;
if(BUTTON_SELECT)
{
    ENABLE_MOVE = !(ENABLE_MOVE);
    if(prerus == 0)
    {
        prerus = 1;
        ENABLE_MOVE = 1;
    }
    def_step_mov = 10;
    ETIMSK=(0<<TICIE3) | (0<<OCIE3A) | (0<<OCIE3B) | (1<<TOIE3) | (0<<OCIE3C) |
(0<<OCIE1C);
    while(BUTTON_SELECT);
}

if(BUTTON_ONOFF)
{
    ENABLE_ROT = !(ENABLE_ROT);
    def_step_rot = 3000;
    ETIMSK=(0<<TICIE3) | (0<<OCIE3A) | (0<<OCIE3B) | (1<<TOIE3) | (0<<OCIE3C) |
(0<<OCIE1C);
    while(BUTTON_ONOFF);
}

if(BUTTON_LEFT)
{
    if(LIMIT_SWITCH_START != 0)
        ETIMSK=(0<<TICIE3) | (0<<OCIE3A) | (0<<OCIE3B) | (1<<TOIE3) | (0<<OCIE3C) |
(0<<OCIE1C);
    else
        ETIMSK=(0<<TICIE3) | (0<<OCIE3A) | (0<<OCIE3B) | (0<<TOIE3) | (0<<OCIE3C) |
(0<<OCIE1C);
}

if(BUTTON_RIGHT)
{
    if(LIMIT_SWITCH_END != 0)
        ETIMSK=(0<<TICIE3) | (0<<OCIE3A) | (0<<OCIE3B) | (1<<TOIE3) | (0<<OCIE3C) |

```

```

(0<<OCIE1C);
    else
        ETIMSK=(0<<TICIE3) | (0<<OCIE3A) | (0<<OCIE3B) | (0<<TOIE3) | (0<<OCIE3C) |
(0<<OCIE1C);
    }

    /*
    //POSUN_VOZÍKU
    ENABLE_MOVE = 1;
    STEP_MOVE = 1;
    DIRECTION_MOVE = 1;
    delay_ms(1);
    ENABLE_MOVE = 1;
    STEP_MOVE = 0;
    DIRECTION_MOVE = 0;
    delay_ms(1);
    */

    /*
    ENABLE_ROT = 0;
    STEP_ROT = 1;
    DIRECTION_ROT = 1;
    delay_ms(100);
    ENABLE_ROT = 0;
    STEP_ROT = 0;
    DIRECTION_ROT = 0;
    delay_ms(100);
    */
    }
}

void numToStr(unsigned char num, unsigned char* str) //Řešení posunu času po 60 (sekundy, minuty, hodiny)
{
    unsigned char number = 0;

    if(num < 10)
    {
        *(str + 0) = '0';
        *(str + 1) = '0' + num;
    }
    else
    {
        *(str + 1) = '0' + (num % 10);
        while(num > 9)
        {
            num -= 10;
            number++;
        }

        *(str + 0) = '0' + number;
    }

    *(str + 2) = '\0';
}

```