



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA STROJNÍHO INŽENÝRSTVÍ  
ÚSTAV MECHANIKY TELES, MECHATRONIKY A  
BIOMECHANIKY**

FACULTY OF MECHANICAL ENGINEERING  
INSTITUTE OF SOLID MECHANICS, MECHATRONICS AND  
BIOMECHANICS

# **NÁVRH POLOHOVACÍHO ZARÍZENÍ MALÉ KAMERY**

DESIGN OF THE POSITIONING DEVICE FOR SMALL CAMERAS

**BAKALÁŘSKÁ PRÁCE**  
BACHELOR'S THESIS

**AUTOR PRÁCE**  
AUTHOR

**LADISLAV MAGYERKA**

**VEDOUCÍ PRÁCE**  
SUPERVISOR

**Ing. ONDŘEJ ANDRŠ, Ph.D.**

BRNO 2015



Vysoké učení technické v Brně, Fakulta strojního inženýrství

Ústav mechaniky těles, mechatroniky a biomechaniky

Akademický rok: 2014/2015

## **ZADÁNÍ BAKALÁŘSKÉ PRÁCE**

student(ka): Ladislav Magyerka

který/která studuje v **bakalářském studijním programu**

obor: **Mechatronika (3906R001)**

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma bakalářské práce:

### **Návrh polohovacího zařízení malé kamery**

v anglickém jazyce:

### **Design of the positioning device for small cameras**

Stručná charakteristika problematiky úkolu:

Cílem této práce bude provést návrh a případnou realizaci polohovacího zařízení pro malé kamery. Předpokládané zařízení bude umožňovat natáčení kamery ve třech osách. Jako pohony budou použity modelářské servopohony. Pro řízení zařízení bude využit vhodný mikrořadič s komunikací do nadřazené vrstvy řízení.

Cíle bakalářské práce:

1. Rešeršní studie malých polohovacích zařízení pro kamery.
2. Návrh mechanické konstrukce polohovacího zařízení.
3. Návrh obslužné elektroniky polohovacího zařízení.
4. Návrh obslužného software.

Seznam odborné literatury:

Havlíček Josef, Vlach Jaroslav, Vlach Martin, Vlachová Viktorie: Začínáme s LabVIEW, BEN - technická literatura, 2008, ISBN / EAN 978-80-7300-245-9 / 9788073002459

Mann Burkhard: C pro mikrokontroléry, BEN - technická literatura, ISBN / EAN 80-7300-077-6 / 9788073000776

Vedoucí bakalářské práce: Ing. Ondřej Andrš, Ph.D.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2014/2015.

V Brně, dne 12.11.2014

L.S.

---

prof. Ing. Jindřich Petruška, CSc.  
Ředitel ústavu

---

doc. Ing. Jaroslav Katolický, Ph.D.  
Děkan fakulty

## **ABSTRAKT**

Cieľom bakalárskej práce bolo navrhnuť zariadenie na polohovanie malej kamery. Prvá kapitola je zameraná na prehľad súčasných dostupných zariadení a popis hlavných použitých prvkov pri návrhu. V ďalších kapitolách sa popisujú vlastnosti mechanizmu, elektroniky a obslužného softvéru.

### **Kľúčové slová**

pan-tilt-roll, PTR, servo, servomotor, kamera, mikrokontrolér, MCU, PWM, C#

## **ABSTRACT**

The aim of this thesis was to design an apparatus for positioning a small camera. The first chapter is focused to an overview of current available devices and description of the main components used during design. In subsequent chapters the features of the mechanism, electronics and utility software are described.

### **Key words**

pan-tilt-roll, PTR, servo, servomotor, camera, microcontroller, MCU, PWM, C#

## **BIBLIOGRAFICKÁ CITÁCIA**

MAGYERKA, Ladislav. Návrh polohovacího zařízení malé kamery. Vysoké učení technické v Brně, Fakulta strojního inženýrství, Ústav mechaniky těles, mechatroniky a biomechaniky, 2015. 36 s. Vedoucí práce Ing. Ondřej Andrš, Ph.D..

## ČESTNÉ PREHLÁSENIE

Prehlasujem, že som bakalársku prácu na tému Návrh polohovacího zařízení malé kamery vypracoval samostatne s použitím odbornej literatúry a prameňov, uvedených na zozname, ktorý tvorí prílohu tejto práce.

---

Dátum

---

Ladislav Magyerka

## **POĎAKOVANIE**

Rád by som poďakoval vedúcemu práce Ing. Ondřejovi Andršovi, Ph.D. za cenné pripomienky a rady pri vypracovaní tejto bakalárskej práce.

Táto práca vznikla za podpory a na prístrojovom vybavení NETME Centre, regionálneho výskumného a vývojového centra vybudovaného z finančných prostriedkov OP VaVpI v rámci projektu NETME Centre (Nové technológie pre strojárstvo), Reg. č.: CZ.1.05/2.1.00/01.0002 a podporovaného v nadväzujúcej fáze udržateľnosti prostredníctvom projektu NETME CENTRE PLUS (LO1202) za finančného príspevku MŠMT v rámci účelovej podpory programu NPU I.

## OBSAH

ÚVOD.....	10
1 PREHĽAD POLOHOVACÍCH ZARIADENÍ KAMIER.....	11
1.1 Typy polohovacích zariadení.....	11
1.2 Servomotory.....	13
1.3 Mikrokontroléry.....	15
2 NÁVRH MECHANICKEJ KONŠTRUKCIE .....	19
2.1 Osi rotácie .....	19
2.2 Konštrukcia.....	19
3 NÁVRH ELEKTRONIKY .....	21
3.1 Nastavenie mikrokontroléru .....	21
3.1.1 Konfiguračné bity.....	21
3.1.2 Časovač.....	22
3.2 Obsluha prerušenia .....	22
3.2.1 Tvorba PWM.....	22
3.2.2 Komunikácia.....	23
4 NÁVRH OBSLUŽNÉHO SOFTVÉRU.....	25
4.1 Užívateľské rozhranie .....	25
4.2 Komunikácia s MCU .....	27
4.3 Obraz z kamery .....	27
ZÁVER.....	28
ZOZNAM POUŽITÝCH ZDROJOV .....	29
ZOZNAM POUŽITÝCH SKRATIEK.....	31
ZOZNAM OBRÁZKOV A TABULIEK .....	32
ZOZNAM PRÍLOH.....	33

## ÚVOD

V dnešnej dobe lacnej elektroniky čím ďalej, tým viac ľudí montuje malé kamery na svoje lietajúce modely na diaľkové ovládanie a roboty. Pevne pripojená kamera niekedy nestačí, preto je nutné ju umiestniť na špeciálny mechanizmus, ktorým je možné ju natáčať do rôznych smerov počas letu alebo iných činností. Bežný užívateľ má k dispozícii ohromné množstvo zariadení, z ktorých každé je určené na iné aplikácie. Niekomu stačí len miniatúrny obrazový snímač na zistenie polohy a vyhýbanie sa prekážkam, niekto zas potrebuje vytvárať kvalitné panoramatické fotografie krajiny z výšky. Bezpečnejším a elegantnejším riešením je natáčať len kameru a nie celý stroj.

V malých diaľkovo ovládaných lietadlách a helikoptéroch sa čoraz častejšie vyskytujú mechanizmy s aktívnou stabilizáciou za pomoci akcelerometrov. Ďalšou novinkou sú veľmi jednoduché a lacné zariadenia schopné rozoznať a sledovať ľudskú tvár alebo predmety s odlišnou farbou, akú má okolie. Nie vždy sú ale predávané v jednom kuse. Často si užívateľ musí sám zaobstarať jednotlivé komponenty a postupovať podľa pokynov pri výrobe. S týmto prístupom si každý môže upraviť a vylepšiť akúkoľvek časť sústavy na základe dostupných materiálov alebo individuálnych potrieb, čo je spolu s nižšími nákladmi na realizáciu veľká výhoda. K tomu je ale potrebné mať požadované vedomosti v danej oblasti.

V tejto práci sú popísané metódy postupu a návrhu všetkých hlavných častí polohovacieho zariadenia s pripomienkami na dôležité vlastnosti, nastavenia a funkcie.

# 1 PREHĽAD POLOHOVACÍCH ZARIADENÍ KAMIER

Kamery samy o sebe nedokážu vykonávať žiadne pohyby, či už translačné, alebo rotačné, pretože neobsahujú patričné aktuátory. Na tieto činnosti slúžia mechanizmy s pohonovými jednotkami, najčastejšie s DC motormi na kontinuálnu rotáciu a servomotormi na presné polohovanie v určitom dovolenom rozmedzí. Každý pohon sprostredkuje rotáciu alebo posunutie práve na jednej ose. Väčšinou sa ale jedná o rotačný pohyb. Motory musia byť nejakým spôsobom riadené. Túto funkciu zvládajú mikrokontroléry, ktoré sú cenovo a aj funkčne vhodné pre takéto jednoduché úkony. Výber správneho čipu a elektroniky závisí na type pohonov, ich vlastností a požadovanom rozlíšení a presnosti polohovania.

## 1.1 Typy polohovacích zariadení

PTZ kamery (pan-tilt-zoom) umožňujú natáčanie v dvoch osách a približovanie pomocou prídavných objektívov alebo digitálne, orezaním a zväčšením obrazu. Najčastejšie sú to bezpečnostné kamery, kde vysoké rozlíšenie a približovanie je nevyhnutnou vlastnosťou. Existuje mnoho rôznych druhov, ktoré sa v zásade líšia hlavne spôsobom pripevnenia na steny a predmety. Najznámejšími z nich sú dome kamery a otočné kamery zobrazené na Obr. 1.1. Dome kamery s kupolovitým tvarom určené pre montáž na strop sú malé, nenápadné a v prevedení antivandal aj odolné voči útokom. Otočné kamery tvaru krabice alebo valca umiestnené najčastejšie na stenách sú síce väčšie a výraznejšie, ale disponujú objektívom, ktorým v spojení s digitálnym priblížením je možné dosiahnuť veľmi vysoké celkové priblíženie [1].



Obr. 1.1: Typy kamier [2]. a) dome kamera, b) otočná kamera.

Mnoho polohovacích zariadení nachádza uplatnenie aj v malých modelárskych lietadlách a vrtuľníkoch najmä kvôli potrebe zachytávať snímky alebo video z výšky, prípadne počas letu. Tieto mechanizmy sú zväčša vyrobené z plastu a hliníku, aby ich váha príliš nenarušila rozloženie hmotnosti. Na internete je možné nájsť neskutočne veľké množstvo najrôznejších typov zariadení, či už v online predajniach ako hotové výrobky, alebo v príspevkoch jednotlivcov na fórach a blogoch, ktoré popisujú postup na vlastnú výrobu a montáž. Často obsahujú aj pokyny na zapojenie elektroniky a ukážky kódu z programu na riadenie pohonov.

Pri výbere správneho mechanizmu je treba zvážiť niekoľko dôležitých faktorov. Hmotnosť a miesto uloženia sú dôležité hlavne pri lietajúcich strojoch, kde každý ušetrený gram znamená viac času stráveného vo vzduchu. Rozmery sa vo väčšine prípadov nedajú príliš zmeniť bez narušenia rovnováhy alebo funkčnosti a je nutné si vybrať iný typ

zariadenia. Prichytenie kamery sa realizuje buď skrutkami, alebo presne na mieru vyrobenými držiakmi. Nie každú kameru je ale možné týmto spôsobom pripevniť. V niektorých menej náročných aplikáciách sa používajú plastové pásky alebo lepidlo.

Alternatívou pre širokú škálu rozličných kamier sú CMUcam(pozri Obr. 1.2) kamery. CMUcam je v podstate jednoduchý obrazový snímač na integrovanom obvode s mikrokontrolérom, ktorý dokáže spracovať obraz, ale aj sledovať farebnú škvrnu. Svojou plochou zadnou stranou je možné ju pripevniť priamo na stojan pomocou skrutiek. Kvôli malým rozmerom sa hodí hlavne na malé roboty [3].



Obr. 1.2: CMUcam kamera [3].

Najjednoduchšie typy polohovacích zariadení obsahujú jedine malé tenké plechy alebo plastové rámy pripevnené priamo na servomotory. Bez použitia ložísk tieto mechanizmy nie je vhodné príliš zaťažovať, pretože hrozí poškodenie ich pohonov. Na Obr. 1.3 je znázornené jedno takéto zariadenie na malom diaľkovo ovládanom aute.



Obr. 1.3: Mechanizmus z plechov [4].

Servomotory sa niekedy montujú z vnútornej strany, aby sa ušetrilo čo najviac miesta. Jeden takýto príklad vidno na Obr. 1.4. Krabíčka motora je v tomto prípade pohyblivá časť a výstupný hriadeľ ostáva stacionárny vzhľadom na súčiastku, s ktorou je spojený. Zvyšná časť mechanizmu je primontovaná na krabíčku skrutkami. To ale môže viesť k situácii, kde je kamera vyosená, čo nemusí byť problém pri sledovaní vzdialených predmetov.



Obr. 1.4: Držiak kamery Fat Shark [5].

Ďalšia často sa vyskytujúca konštrukcia (pozri Obr. 1.5) je omnoho pevnejšia, veľaokrát podporená aj ložiskami, a dokáže uniesť bežné digitálne kamery. Metóda pripevnenia sa líši na voľbe konkrétneho modelu kamery, ale zvyčajne nie je potrebný ďalší držiak.



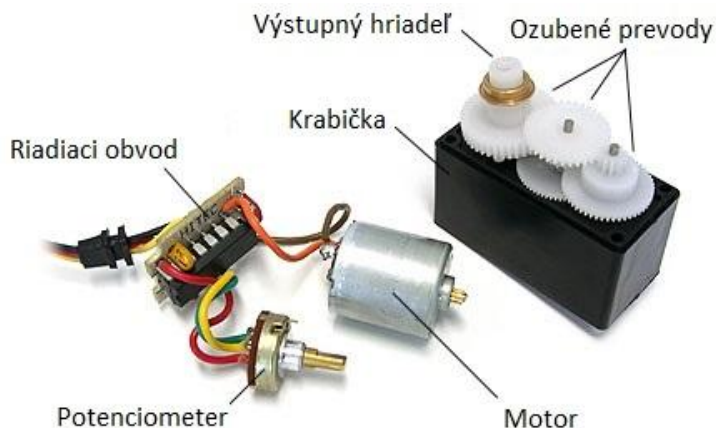
Obr. 1.5: Model mechanizmu PT-3 [6].

## 1.2 Servomotory

Hlavným prvkom elektronicky riadených polohovacích mechanizmov malých kamier sú vo väčšine prípadov modelárske servomotory. Ich malé rozmery, jednoduchosť ovládania a nízka cena z nich robia jednoznačnú voľbu na použitie v podobných triviálnych zariadeniach. Navyše je možné si vybrať vhodný servomotor podľa požadovaných rozmerov, rýchlosti otáčania a momentu zo širokej škály modelov.

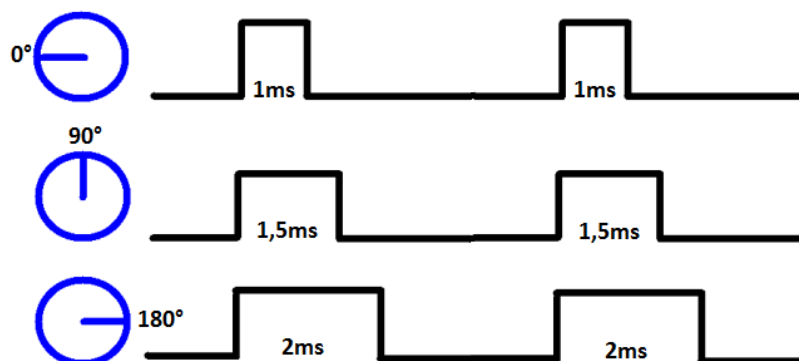
Servomotor je malé zariadenie obsahujúce jednosmerný motor, ozubené prevody, potenciometer, integrovanú elektroniku a výstupný hriadeľ, ako je vidno na Obr. 1.6. Potenciometer je spojený s výstupným hriadeľom a pôsobí ako zariadenie na spätnú väzbu. V závislosti od uhlu natočenia a vstupného signálu sa snaží dostať hriadeľ do správnej pozície. Prevody sú väčšinou vyrobené z nylonu, plastu alebo z rôznych druhov kovov,

ale výkonnejšie a kvalitnejšie servomotory majú prevody z titánu, ktorý je mnohonásobne pevnejší a odolnejší na opotrebovanie [7,8].



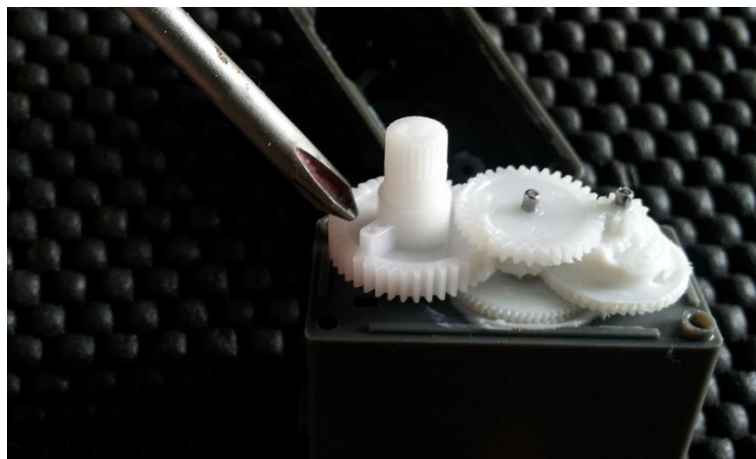
Obr. 1.6: Komponenty servomotora [8].

Napájanie servomotora sa rieši tromi vodičmi, dve na jednosmerné napätie v rozmedzí 4 V až 6 V a jeden na riadiaci signál, ktorý sa vytvorí pulznou šírkovou moduláciou. Frekvencia jednotlivých pulzov pre analógové servomotory musí byť 50 Hz, čiže jeden pulz každých 20 milisekúnd (ms), a pre digitálne 300 Hz. Šírka pulzov rozhoduje o tom, na ktorú stranu má motor točiť. Neutrálna poloha je vždy v okolí 1,5 ms, no jednotlivé servomotory majú rôzne hraničné hodnoty. Väčšinou sa ale pohybujú v rozmedzí približne 1 ms až 2 ms, čo je znázornené na Obr. 1.7.



Obr. 1.7: Natočenie motora v závislosti na šírke pulzu [9].

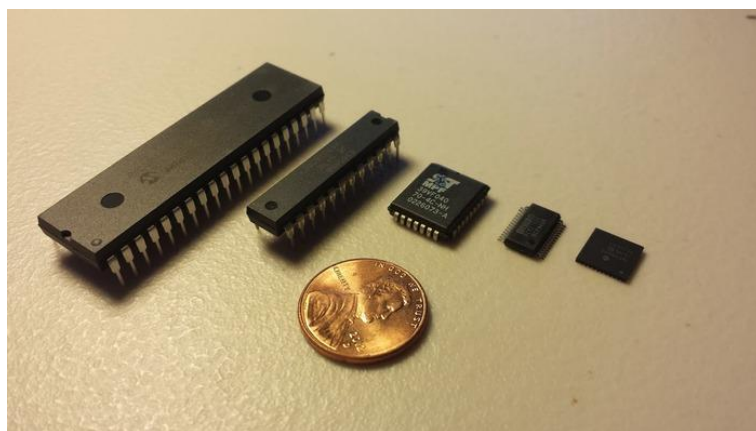
Motor sa nedokáže otočiť na viac ako jeho stanovený rozsah stupňov kvôli malému výbežku na jednom z prevodov (pozri Obr. 1.8), ktorý zabráni ďalšej rotácii. Pri niektorých typoch servomotorov je ale možné odstrániť tento výbežok a nahradiť potenciometer dvomi rovnakými rezistormi s ekvivalentným odporom v sérii. Po tejto úprave sa riadiacim signálom namiesto polohy udáva rýchlosť a motor sa dokáže kontinuálne otáčať s vyšším výkonom ako pri jednosmerných motoroch podobnej veľkosti [10].



Obr. 1.8: Výbežok na zabránenie pretočenia [11].

### 1.3 Mikrokontroléry

Mikrokontrolér (MCU) je malý počítač na jednom integrovanom obvode, ktorý obsahuje procesor, pamäť RAM, programovateľnú pamäť ROM alebo flash, vstupno-výstupné periférie a oscilátor. Vo väčšine prípadov však zahŕňa aj pokročilejšie prvky, ako napríklad časovače, A/D a D/A prevodníky, komparátory, PWM moduly a obvody na sériovú a paralelnú komunikáciu. Všetky tieto prvky sú navrhnuté na mieru a sústredené na veľmi malé miesto. Veľkosťou sú podobné ostatným bežne používaným prvkom v elektrických obvodoch, a preto nie je ich zakomponovanie do väčšiny obvodov problematické. Na Obr. 1.9 je znázornené porovnanie veľkostí rôznych mikrokontrolérov s mincou.

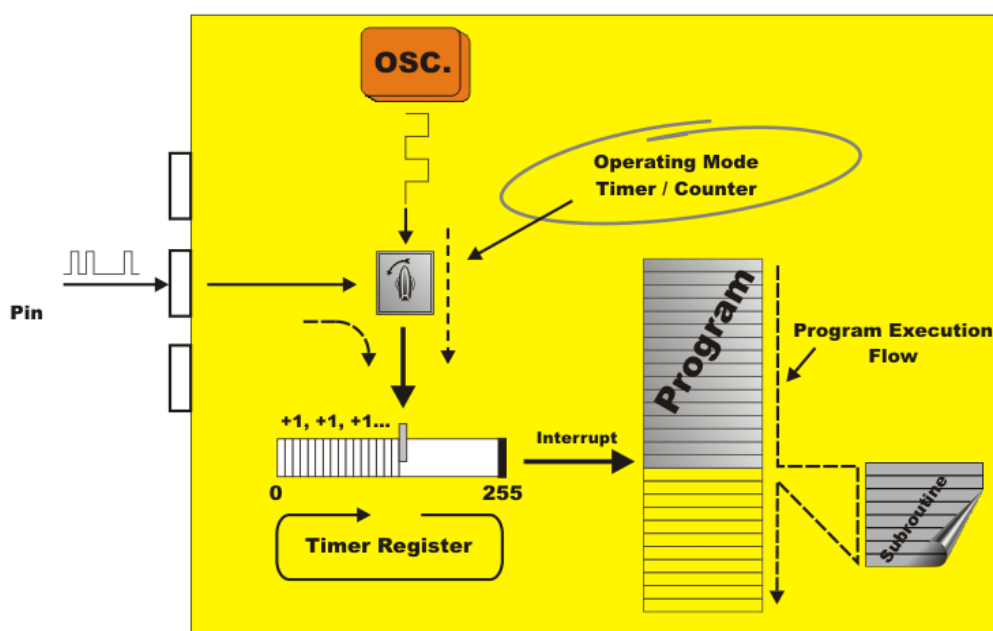


Obr. 1.9: Rôzne veľkosti mikrokontrolérov [12].

Mikroprocesory sú samostatné procesné jednotky, ktoré na svoju činnosť potrebujú ďalšie zapojené komponenty. Tie sa môžu zamieňať za iné typy s rôznymi parametrami. Týmto spôsobom sa dá poskladať ľubovoľná modulárna zostava s požadovanými funkciami a vlastnosťami.

Operačné kmitočty väčšiny bežných mikrokontrolérov sa pohybujú v jednotkách až desiatkach MHz a pri vysoko výkonných modeloch dosahujú aj stovky MHz. Užívateľ má možnosť pridaním externého oscilátora zvýšiť alebo znížiť operačný kmitočet, ale iba v uvedenom rozsahu, na ktorý je daný model navrhnutý. V niektorých prípadoch ich je možné nastaviť na oveľa nižšie frekvencie, ktoré dosahujú až 16 kHz, pri požiadavkách na nízku spotrebu energie [13,14].

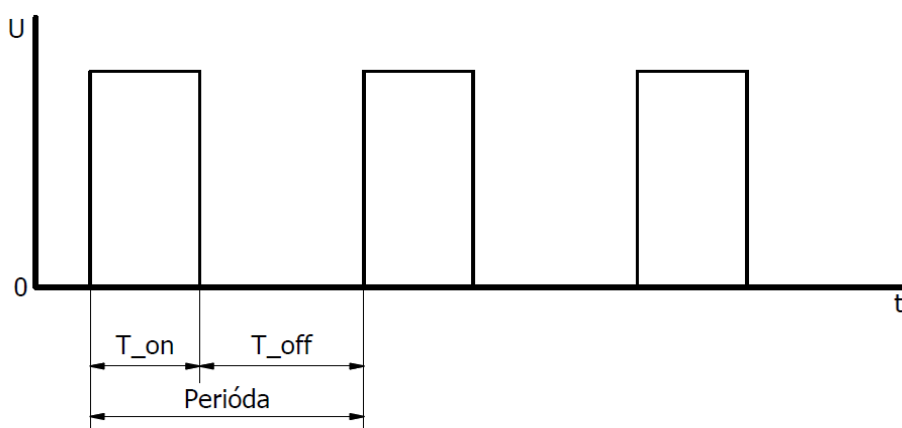
Časovač (timer) je v podstate počítadlo (counter), ktoré v dnešnej dobe poskytuje takmer každý mikrokontrolér. Väčšina z nich však podporuje aj viac ako jeden časovač s rozlíšením 8 bitov až 16 bitov a v niektorých prípadoch aj s vyšším. Najčastejšie sa využívajú na meranie času medzi určitými externými udalosťami alebo na pravidelné prerušovanie programu. V prípade neprítomnosti PWM modulu alebo jeho nízkeho bitového rozlíšenia na čípe, je časovač vyhovujúcou a prakticky jedinou ekvivalentnou alternatívou na dosiahnutie potrebného obdĺžnikového signálu s konštantnou periódou bez dodatočných hardvérových prvkov. V závislosti na operačnom kmitočte procesoru, deliči (prescaler) a rozlíšení sa dá získať široký rozsah frekvencií, na ktoré už napríklad 10-bitový PWM modul nedokáže byť nastavený.



Počítadlo sa v každom tiku hodín procesoru buď inkrementuje, alebo dekrementuje, v závislosti na zvolenom smere. Po pretečení tejto hodnoty sa vyvolá prerušenie a následne sa vykonajú príkazy príslušného podprogramu obsluhy tohto prerušenia. Tento cyklický dej je znázornený na Obr. 1.10. K aktuálnej hodnote sa pristupuje cez špeciálne funkčné registre (SFR), v ktorých je priamo uložená, či už ako jedno 16-bitové, alebo dve 8-bitové čísla, ktoré sa nazývajú horný a dolný byte. Problémy s presnosťami môžu nastať pri čítaní a zapisovaní dvoch čísel počítadla za sebou, pretože po zmene alebo uložení prvého čísla a následnom prístupe k druhému číslu, sa prvé číslo mohlo medzitým zmeniť. Toto sa rieši

pridaním registra na uchovanie horného bytu. Pri čítaní dolného bytu sa horný byte súčasne uloží do zásobníku (buffer) v registri, z ktorého sa potom prečíta v nasledujúcej inštrukcii. Pri zápise horného bytu sa táto hodnota uloží do zásobníku a až pri zápise dolného bytu sa obe hodnoty naraz zapíšu do registra [15].

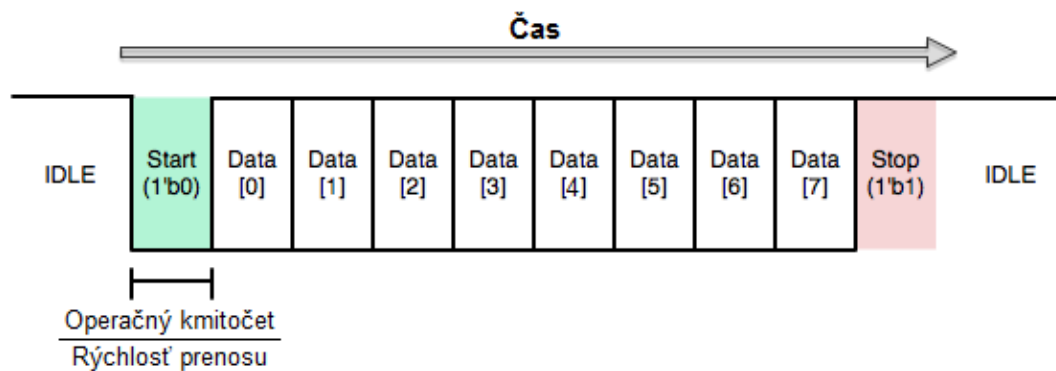
Pulzná šírková modulácia (PWM) je metóda prenosu analógového signálu s dvomi hodnotami. Hlavné využitie má vo výkonovej elektronike na reguláciu energie dodávanej do elektrických zariadení, hlavne motorov. Veľmi rýchlym zapínaním a vypínaním napájania zo zdroja konštantného napätia je možné meniť strednú hodnotu napätia záťaže, ktorá je priamo úmerná striede, čiže pomeru doby v zapnutom stave k celkovej perióde. Na Obr. 1.11 je znázornená závislosť napätia na čase diagramu PWM [17].



Obr. 1.11: PWM diagram.

UART (universal asynchronous receiver/transmitter) je modul schopný obojsmernej asynchrónnej sériovej komunikácie práve medzi dvomi zariadeniami pomocou dvoch vodičov. Jeden na vysielanie (transmit) a druhý na príjem (receive). Samozrejme, je možná aj jednosmerná komunikácia. Tieto moduly obsahujú vlastný generátor hodín a registre na dočasné uloženie prijímaných a odosielaných údajov. Taktiež umožňujú v programe nakonfigurovať rôzne vlastnosti UARTu. Počet dátových bitov sa nastavuje medzi 5 až 9 bitmi, v závislosti od typu modulu, ale najčastejšie to je 8 bitov, čo v programoch postačuje na jeden znak z ASCII tabuľky. Paritný bit nie je povinný, ale zaisťuje jednoduchú detekciu chýb v komunikácii. Ďalej sa nastavuje počet stop bitov, jeden až dva bity, ktoré signalizujú koniec znaku. Dôležité je správne nastaviť rýchlosť prenosu (baud rate) na oboch komunikujúcich zariadeniach z radu prípustných hodnôt, ktoré sú závislé na operačnom kmitočte jednotlivých zariadení.

V nečinnom stave sú vodiče pod napätím, aby sa vedelo, že nie sú porušené. Prenos znaku začína jedným štart bitom v logickej nule, ďalej pokračujú dátové bity, paritný bit a jeden alebo dva stop bity v logickej jednotke. Na Obr. 1.12 je zobrazený podobný prenos, akurát obsahuje len jeden stop bit a nemá zavedenú paritnú kontrolu. Celá komunikácia prebieha asynchrónne od oboch komunikujúcich zariadení, ktoré musia jedine zapisovať údaje na odoslanie do registra a čítať z neho prijaté údaje, ak sú k dispozícii. UART moduly sa postarajú o všetko ostatné.



Obr. 1.12: Prenos jedného znaku [18].

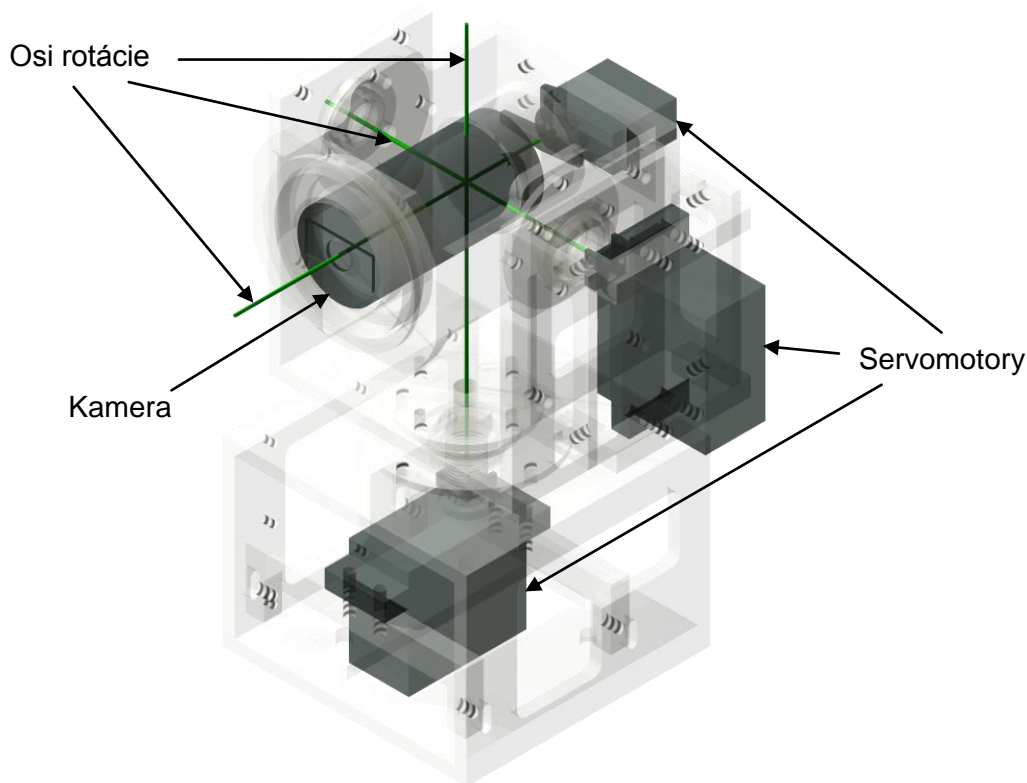
USART (universal synchronous/asynchronous receiver/transmitter) je v podstate vylepšenou verziou UARTu, ktorá dokáže sprostredkovať efektívnejšiu komunikáciu synchronizovaním hodín na oboch moduloch. Odosielajú sa len dátové bity, čiže štart bit, paritný bit a stop bity sú úplne vynechané. Na zachovanie synchronizácie je potrebné vo voľnom čase, teda ak dátové bity nie sú k dispozícii, neustále odosielať znak SYN, čo ale moduly vykonávajú automaticky [15,19].

## 2 NÁVRH MECHANICKEJ KONŠTRUKCIE

Pri návrhu mechanickej konštrukcie bolo potrebné dodržať určité požiadavky na funkciu, dostupnosť materiálov a možnosti výroby. Zariadenie muselo byť schopné natáčať kameru v troch osiach použitím servomotorov. Jedná sa teda o PTR (pan-tilt-roll) mechanizmus, ktorého virtuálny model bol vytvorený v programe Autodesk Inventor Professional 2015. Výkres zostavy je dostupný v prílohách.

### 2.1 Osi rotácie

Keďže dodaná web kamera nedisponuje funkciou približovania, používala by sa prevažne na krátke vzdialenosti, kde ale nastáva problém s jej možným vyosením. To znamená, že os obrazového snímača nepretína osi otáčania a akýkoľvek rotačný pohyb zmení aj jeho polohu v priestore. Aby sa tento problém v danom mechanizme nevyskytoval, osi rotácie vystupujúce z hriadeľov servomotorov boli vzhľadom na seba umiestnené tak, že sa všetky tri prekrížovali v jednom bode (pozri Obr. 2.1). Znamky vyosenia kamery z dôvodu neznámej polohy čipu obrazového snímača síce zotrávajú, ale ich efekt je len minimálny a zanedbateľný.



Obr. 2.1: Osi rotácie kamery.

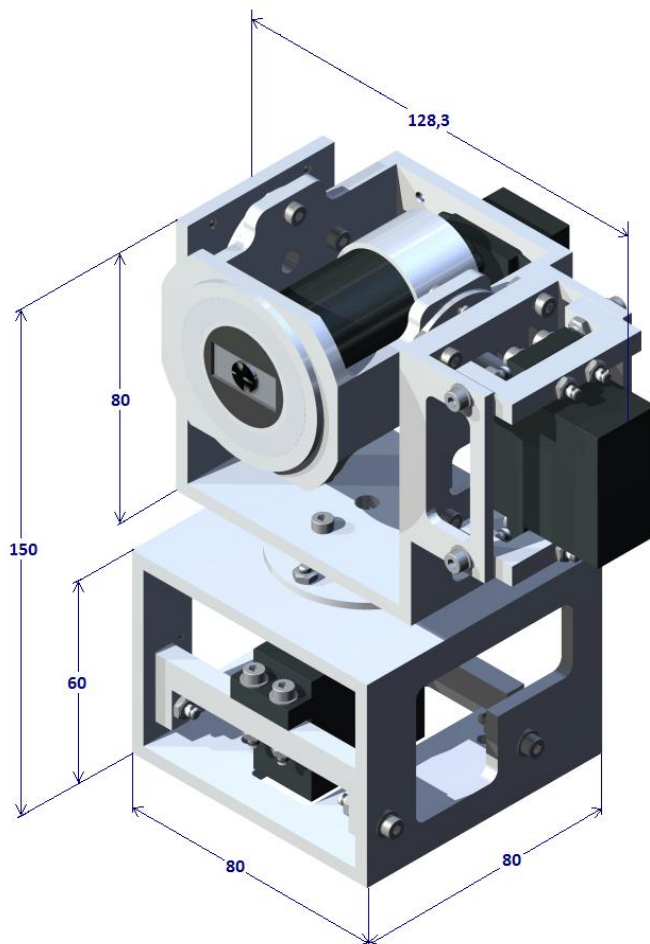
### 2.2 Konštrukcia

Základná konštrukcia sa skladá zo štyroch hliníkových profilov, na ktoré sa montujú všetky ostatné súčiastky. Spodný rám má v sebe jeden servomotor, ktorý otáča s hriadeľom pripevneným k prostrednému rámu. Aby sa prevody motora nepoškodili, na udržanie vrchnej časti mechanizmu sa použilo ložisko 61700 pevne uložené medzi svojim osadením, upínacím puzdrom a maticou, ktorá ho spolu s hriadeľom uchyťí na mieste. Týmto spôsobom sú oba rámy bezpečne spojené aj pri prevrátení celej konštrukcie. Väčšie

ložisko by však bolo na danom mieste vhodnejšie kvôli vyšším momentom, ktoré by naň pôsobili pri naklonení o 90 stupňov.

Druhý rám v strede mechanizmu má k sebe z jednej strany pripevnený servomotor, ktorý pomocou dvoch hriadel'ov v ložiskách 61700 natesno drží a otáča vrchný rám s kamerou. Upínacie puzdra a matice nie sú v tomto prípade nutné, pretože pri vložení všetkých dielov medzi steny tohto profilu je pohyb v smere kolmom na os hriadeľa nemožný. Samotná kamera je z prednej strany vsunutá spolu s distančným krúžkom do ložiska 61806 pripevneného na vrchný rám. Zo zadnej strany kameru obklopuje dutý, valcový držiak prichytený k tretiemu, najmenšiemu servomotoru, napevno spojenému s týmto rámom, čo umožňuje tretí smer otáčania.

Z hliníkových profilov sa z dôvodu odl'ahčenia celej sústavy odobrali nepotrebné časti materiálu. Na dvoch vrchných rámoch sa nachádzajú nevyužívané diery so závitmi, ktoré však môžu slúžiť na dodatočné vyváženie konštrukcie primontovaním ľubovoľných závaží. Virtuálny model mechanizmu je znázornený na Obr. 2.2.



Obr. 2.2: Model mechanizmu.

### 3 NÁVRH ELEKTRONIKY

Za pohonové jednotky sa vybrali tri analógové modelárske servomotory, ktoré už boli kúpené a dostupné. Dva kusy Hitec HS-475HB a jeden menší ModelleXXpert MEX-12. Všetky tri pracujú v rozsahu napätia 4,8 V až 6 V a sú schopné veľkého uhlu natočenia, na čo sa pri návrhu mechanizmu myslelo.

Riadenie servomotorov sa realizovalo mikrokontrolérom PIC16F887 od spoločnosti Microchip, ktorý disponuje všetkými vlastnosťami (pozri Tab. 3.1) potrebnými na splnenie požiadaviek tejto práce. PWM modul je síce kvôli svojmu nedostatočnému 10bitovému rozlíšeniu absolútne nevhodný na riadenie servomotorov, ale správnym nastavením časovača je možné dosiahnuť adekvátnej frekvencie prerušenia a rozlíšenia. Pomocou modulu EUSART sa zaistí komunikácia s iným kompatibilným hardvérom.

Tab. 3.1: Parametre PIC16F887 [20].

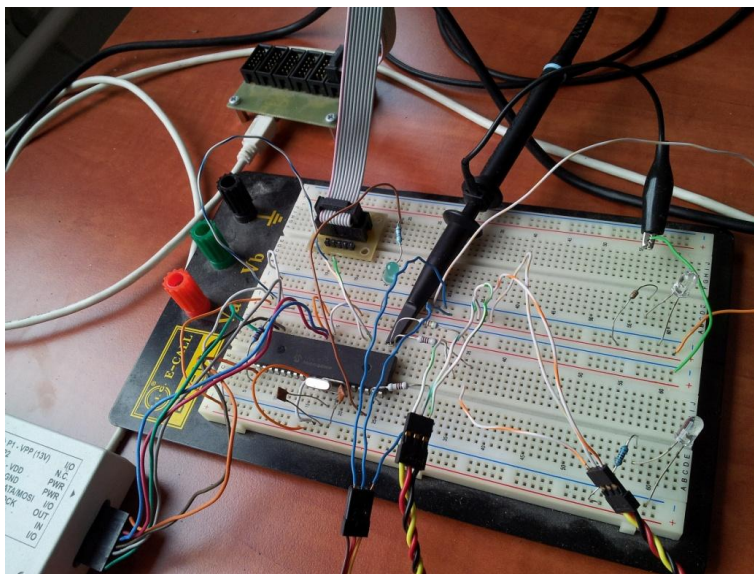
Názov parametru	Hodnota	Názov parametru	Hodnota
Typ Programovej Pamäte	Flash	Časovače	2 x 8-bit, 1 x 16-bit
Programová Pamäť (KB)	14	ADC	14 ch, 10-bit
Rýchlosť CPU (MIPS)	5	Komparátory	2
RAM (byte)	368	Počet Pinov	40
EEPROM (byte)	256	Teplotný Rozsah (C)	-40 to 125
Digitálne Komunikačné Periférie	1-UART, 1-SPI, 1-A/E/USART, 1-I2C1- MSSP(SPI/I2C)	Capture/Compare/PWM Periférie	1 CCP, 1 ECCP
		Operačný napäťový rozsah (V)	2 to 5.5

#### 3.1 Nastavenie mikrokontroléru

Celý kód sa písal v jazyku C v programe MPLAB X IDE v2.26. Na jeho testovanie sa použilo kontaktné pole (breadboard) so zapojenou elektronikou, ako je to zachytené na Obr. 3.1. Mikrokontrolér bol napájaný priamo z programátora ASIX PRESTO a servomotory zase z laboratórneho zdroja napätia. Schéma zapojenia a doska plošných sú dostupné v prílohách.

##### 3.1.1 Konfiguračné bity

Predvolená hodnota operačného kmitočtu 8 MHz sa zmenila na 20 MHz pridaním externého kryštálového oscilátora a prepísaním príslušných konfiguračných bitov. Na zabezpečenie správnej činnosti sa tesne pri oscilátor pridali a zapojili dva kremíkové kondenzátory s kapacitou 22 pF odvedené na zem. Watchdog timer v takomto jednoduchom programe nebol potrebný, a tak sa vypol spolu aj s ochranou kódu. Tieto nastavenia sa nachádzajú v súbore configuration\_bits.c.



Obr. 3.1: Kontaktné pole s elektronikou.

### 3.1.2 Časovač

PWM signál na riadenie servomotorov bolo možné dosiahnuť jedine využitím časovača. Zvolil sa Timer0, s ktorým sa z hľadiska čítania a zapisovania do registra pracuje najjednoduchšie. Delič kmitočtu sa nastavil na najnižší možný, teda 1:2, čo spolu s interným deličom 1:4 medzi frekvenciou oscilátora a inštrukčnými hodinami dáva celkové delenie 1:8. Na dosiahnutie frekvencie prerušenia 100 kHz sa v každej obsluhu tohto prerušenia zadá začiatková hodnota počítadla TMR0 na 231. Bit vlajky pretečenia počítadla T0IF sa taktiež musí v programe vynulovať. Týmto spôsobom by počas 20 ms, čo je požadovaná perióda signálu servomotorov, nastalo 2000 takzvaných krokov, čo sú v podstate prerušenia počas jednej periódy.

Pomocou osciloskopu sa zistilo, že takéto veľké množstvo operácií za daný časový úsek mikroprocesor nezvláda. Požadovaná hodnota krokov sa jednoduchou trojčlenkou vypočítala na 1163. Vďaka pevnému počtu inštrukcií v samotnom prerušení je perióda stabilná. Následne sa zistil počet krokov ekvivalentných dobe 1,5 ms, čo je neutrálna poloha servomotorov, v tomto prípade 85 krokov. Jednotlivé motory majú rôzne limitné hodnoty, do ktorých sa dokážu natočiť, takže ich bolo potrebné experimentom zistiť. Čísla, ktoré nezapadajú do týchto rozmedzí, nie sú použité na zmenu natočenia. Namiesto toho sa premenné nastavujú na hodnotu neutrálnej polohy, čo značí, že nastala chyba v komunikácii alebo v odosielaní údajov.

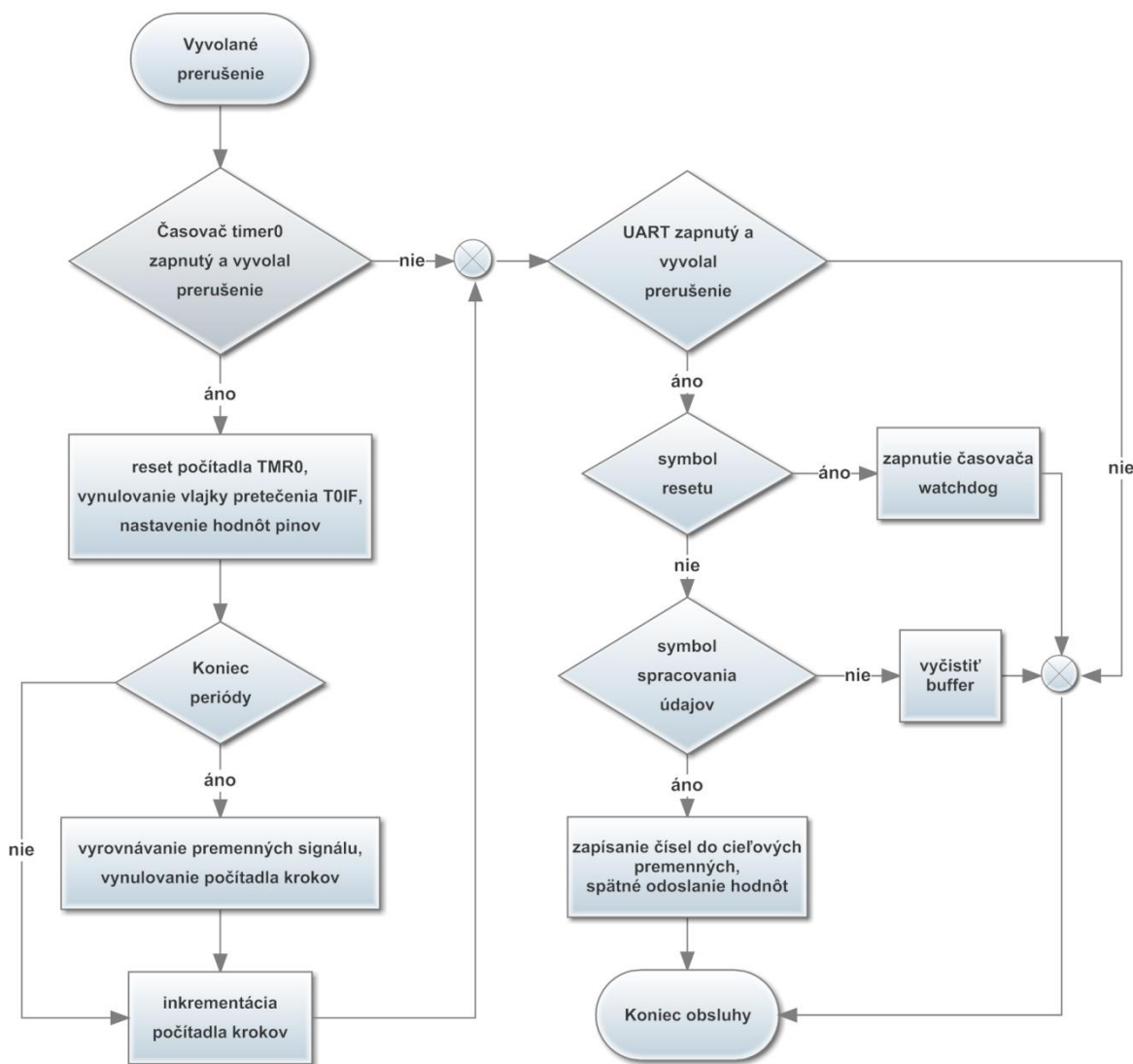
## 3.2 Obsluha prerušenia

Funkcia na obsluhu prerušenia je v tomto konkrétnom mikrokontroléri len jedna. Preto je nutné kontrolovať bity prislúchajúce k jednotlivým aktivovaným modulom a presmerovať chod programu do správneho podprogramu funkcie, ako je znázornené na Obr. 3.2.

### 3.2.1 Tvorba PWM

V podprograme, vyvolanom prerušením od časovača, sa v každom cykle inkrementuje hodnota iteračnej premennej `pwm_steps_counter`. V hodnote 0 sa piny vedúce signál do motorov nastavujú na logickú jednotku. Každý servomotor má v programe dedikované dve premenné na zmenu svojho signálu. Prvá premenná je v každom cykle porovnaná

s aktuálnou hodnotou iteračnej premennej. Pri zhode týchto čísel sa príslušný pin nastaví na logickú nulu, čo znázorňuje koniec pulzu v danej perióde. Do druhej premennej je priamo zapisovaná hodnota získaná z počítača pomocou modulu UART a značí cieľovú hodnotu, na ktorú sa má prvá premenná dostať. Tá sa konci každej periódy, v hodnote 1163, pri nezhodných číslach buď inkrementuje, alebo dekrementuje, až kým sa obe nerovnajú. Tým sa zaistí plynulé natáčanie servomotorov.



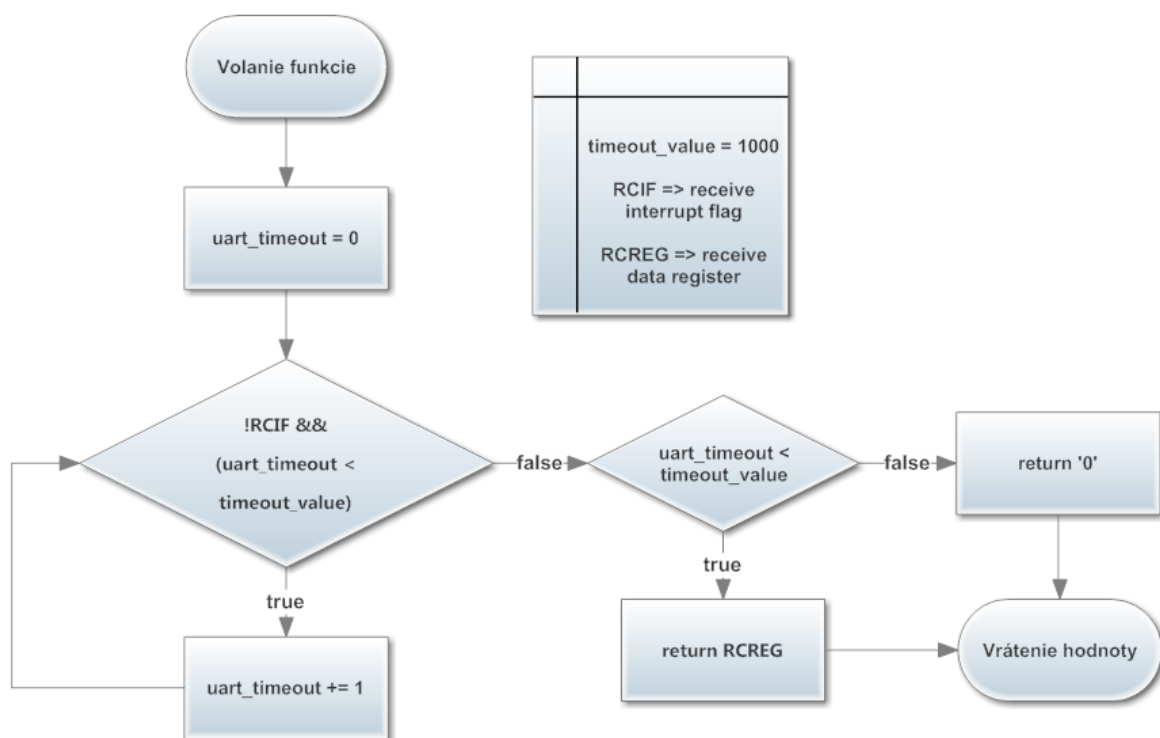
Obr. 3.2: Vývojový diagram obsluhy prerušenia.

### 3.2.2 Komunikácia

Zadávanie povelov do pohonov sa realizuje pomocou modulu EUSART medzi počítačom a mikrokontrolérom. Na samotnú komunikáciu sa používa prevodník USB-UART CP-2102 z dôvodu kompatibility s novšími počítačmi a notebookmi. Nastavená prenosová rýchlosť (baud rate) je 9600 Bd a využíva sa 8bitový prenos v asynchrónnom móde. Po prijatí prvého znaku reťazca sa vyvolá prerušenie a zistí sa, aký je to symbol. Podľa neho sa pomocou funkcie switch prejde do vhodnej vetvy obsluhy. Jedna vetva vedie k manuálnemu resetovaniu čipu zapnutím časovača watchdog, druhá slúži na spracovanie

údajov o požadovanej polohe motorov. Ak prijatý symbol nevyhovuje prvým dvom podmienkam, buffer sa prečistí. Počítač odosiela deväť znakov typu char za sebou, ktoré sa konvertujú na tri čísla a zapíšu do cieľových premenných. Po prevzatí týchto znakov sa pre istotu prečistí buffer a spätne sa odošlú zapísané hodnoty z premenných, aby užívateľ mohol skontrolovať ich správnosť.

Funkcie na obsluhu komunikácie sa postupne rozvíjajú od čítania a zapisovania jedného znaku až po spracovanie reťazcov a čísel, čo v podstate znamená opakovanie základnej funkcie po určitý počet cyklov a ich typovú konverziu. Na ochranu proti zastaveniu chodu programu čakaním na prítomnosť nového znaku na prevzatie sa zaviedol časovač do funkcie, ktorá má na starosť čítanie tohto znaku. Ten pracuje na báze jednoduchého podmieneného cyklu s počítadlom, ktoré je postupne inkrementované až do hodnoty nastavenej na 1000 cyklov. Ak sa očakávaný znak do tej doby neobjaví v registri, funkcia vráti nulu (pozri Obr. 3.3). Číslo 1000 sa zvolilo len odhadom a je možné ho ešte znížiť, ale v danej aplikácii nie je potrebné dokonale vyladiť ochranný prvok, ku ktorému v praxi môže dôjsť jedine pri odpojení vodičov na komunikáciu počas odosielania reťazca znakov do čipu.

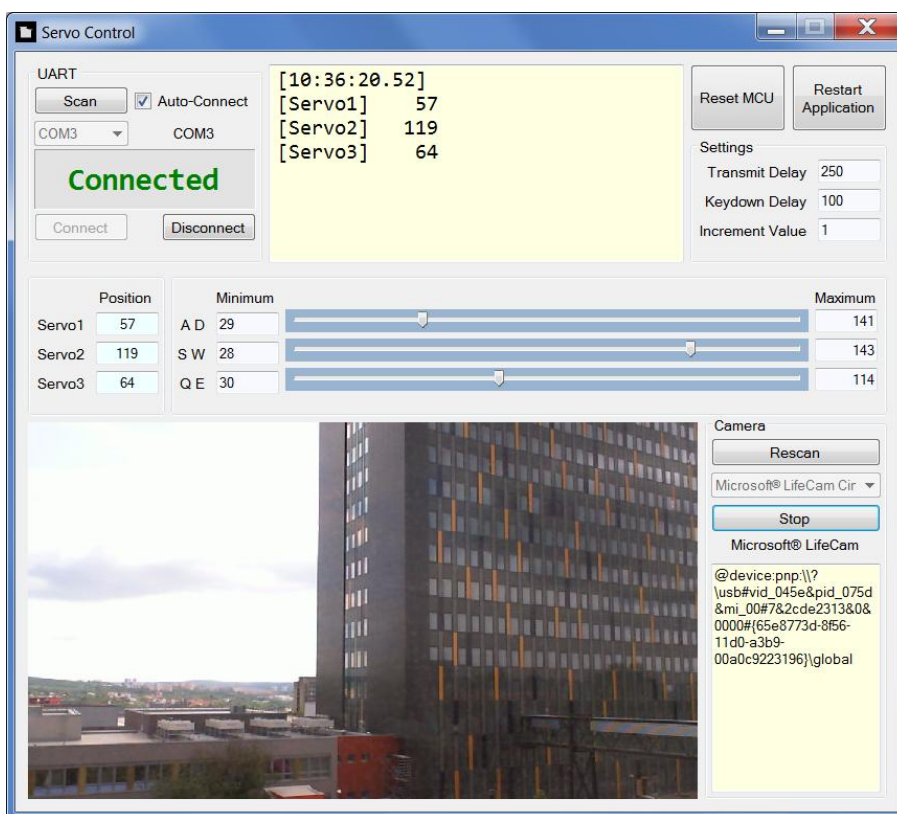


Obr. 3.3: Vývojový diagram funkcie na čítanie jedného znaku.

## 4 NÁVRH OBSLUŽNÉHO SOFTVÉRU

Hlavným dôvodom voľby programovacieho jazyka C# na vytvorenie užívateľského rozhrania (pozri Obr. 4.1) v subsysteme WPF je jednoduchosť práce s grafickými prvkami a prepojeniami medzi nimi. Knižnica System.IO.Ports navyše obsahuje všetky potrebné príkazy a funkcie na bezproblémovú komunikáciu s mikrokontrolérom pomocou UARTu. Na prácu sa využil program Visual Studio Express 2012. Keďže notebooky neobsahujú sériový port, prevodníkom USB-UART sa musel v systéme vytvoriť virtuálny port COM3.

Dôležitým prvým krokom pri vytváraní programu na komunikáciu dvoch zariadení je zaistiť, aby sa v pamäti nenachádzalo viac inštancií určitého procesu, ktoré by mohli spôsobovať chaos. Tomuto problému sa dá jednoducho vyhnúť využitím objektu Mutex (mutual exclusion) v súbore Program.cs. Pri pokuse o spustenie programu sa overí, či sa už v systéme vyskytuje mutex s rovnakým identifikačným reťazcom (ID). Ak áno, objaví sa varovné hlásenie a nový proces sa ihneď ukončí, inak sa program normálne spustí a vytvorí a privlastní si mutex so zadaným ID, ktorý ostane v systéme, až kým sa daná aplikácia nevykone.



Obr. 4.1: Hlavné okno programu.

### 4.1 Užívateľské rozhranie

Na zmenu polohy mechanizmu sa do mikrokontroléru musia odoslať tri čísla, ktorým predchádza jeden symbol označujúci začiatok ich zápisu do patričných premenných. Tieto čísla je možné zmeniť dvoma spôsobmi. Prvým je priame zapisovanie určitej hodnoty do textového poľa a druhým je použitie ovládacieho prvku TrackBar, ktorý je tiež nazývaný ako Slider. Ten umožňuje plynule meniť jeho hodnotu posúvaním jazdca, či už myšou, alebo šiestimi tlačidlami (W A S D Q E), dve na každú jednu os rotácie.

TrackBar v základnom nastavení nie je pre túto činnosť veľmi užitočný, pretože kliknutím na ktorékoľvek miesto na lište posunie kurzor o pevný počet tikov. Pridaním obsluhy udalosti MouseDown, ktorá nastane po kliknutí na lištu, sa nastaví hodnota prislúchajúca aktuálnej pozícii myši vzhľadom na daný ovládací prvok. Aby bolo možné súčasne ovládať všetky tri kurzory, udalosťou KeyDown hlavného okna programu sa podľa stlačených kláves nastaví príslušné logické premenné na „true“ a spustí sa časovač timerKeyDownDelay, ktorý pri každom prerušení na základe týchto premenných zvýši alebo zníži hodnotu TrackBarov. Po pustení klávesy sa v obsluhu KeyUp daná logická premenná vynuluje a ak sú všetky nulové, časovač sa vypne. Akákoľvek zmena v textovom poli taktiež vyvolá zmenu hodnoty vedľajšieho TrackBaru. To platí, samozrejme, aj naopak. Deje sa to aj po opustení daného ovládacieho prvku alebo vyvolaní funkciou Invoke z iných častí programu. Aby sa predišlo zapisovaniu iných znakov ako čísel, každé stlačenie klávesy v textových poliach sa overí a nevhodné znaky sa ignorujú. Navyše pred schválením a používaním akéhokoľvek zadaneého čísla sa musí najprv skontrolovať, či sa nachádza v povolenom rozsahu. Ak nie, nastaví sa na svoju hraničnú hodnotu. Na bokoch každého TrackBaru sú dve textové polia, ktoré umožňujú meniť jeho minimum a maximum, a teda aj premennej udávajúcej natočenie mechanizmu. Týmto spôsobom sa dajú nastaviť vlastné limity, ktoré ale majú aj svoje vlastné limity kvôli zabráneniu poškodenia servomotorov.

V pravom hornom rohu sa nachádzajú dve tlačidlá, jeden na reštartovanie aplikácie, druhý na softvérový reset mikrokontroléru, ktorý je ale možný len počas komunikácie s ním. V kolónke Settings sa nastavujú periódy dvoch časovačov a konštanta, ktorá inkrementuje alebo dekrementuje hodnotu TrackBaru.

V ľavom hornom rohu sú ovládacie prvky podieľajúce sa na komunikácii pomocou UARTu. Tlačidlom Scan sa vyhľadajú všetky sériové porty, zapíšu sa do zoznamu pod ním a v prípade, že sa nenájdu žiadne, varovné hlásenie sa vypíše do veľkého textového poľa napravo. Ak je aspoň jeden port prístupný, v zozname sa označí prvý z nich a odblokuje sa tlačidlo Connect, ktorým sa inicializuje protokol UART. Keďže zlyhané spojenie automaticky nepreruší komunikáciu, zaškrtnutím poľa Auto-Connect sa periodicky každú sekundu kontroluje, či je používaný port stále aktívny. Odpojením kábla od prevodníku sa daný port v systéme už nezobrazí, takže komunikácia sa preruší. Aj v tomto stave je naposledy používaný port periodicky vyhľadávaný kvôli opätovnému obnoveniu pripojenia k mikrokontroléru. Tlačidlá Connect a Disconnect nikdy nemôžu byť aktívne naraz, aby sa predišlo zastaveniu programu. Aktivovaním jedného sa následne deaktivuje druhé.

Dolná časť hlavného okna obsahuje obrazové okno, do ktorého sa vykresľujú snímky z kamery. Jeho rozmery sú 640x360 pixlov, čo je presne polovica rozlíšenia tejto kamery. Tlačidlom Rescan sa, podobne ako pri UARTE, vyhľadajú a uložia názvy všetkých video zariadení do zoznamu, ktoré označí buď prvú položku, alebo keď už jedno zariadenie bolo používané, tak má prednosť. Na spustenie a zastavenie zobrazovania videa z kamery je použité iba jedno tlačidlo (Start/Stop), ktoré akurát mení svoj text podľa stavu prehrávania. Pod ním sa zobrazí názov aktuálneho zariadenia a aj jeho zdroj, ktorý je síce pre bežného užívateľa nepotrebný, ale pri práci s viacerými rovnakými kamerami ich pomáha lepšie identifikovať. Po strate signálu sa prehrávanie automaticky zastaví, ale už sa znova spustiť nepokúša.

## 4.2 Komunikácia s MCU

Nastavenia UARTu sú identické ako v mikrokontroléri, čiže 8 dátových bitov na jeden znak pri prenosovej rýchlosti 9600 Bd v asynchrónnom móde. Po príchode znaku sa spustí funkcia na obsluhu tejto udalosti, ktorá prečíta a uloží všetky znaky do reťazca, až kým nenarazí na indikátor nového riadku („\r\n“). Následne sa volá ďalšia funkcia na spracovanie a vypisovanie údajov. Mikrokontrolér odosiela tri reťazce za sebou, z ktorých každý obsahuje jeden identifikačný symbol, tri číslice a dva znaky nového riadku. Z reťazca sa symbol vymaže, namiesto neho sa vloží označenie servomotora, ku ktorému nasledovné číslo patrí, a nakoniec sa všetko vypíše do textového poľa.

Čítanie údajov týmto spôsobom môže vyvolať výnimku `TimeoutException` v prípade, že sa do určitej doby znak nového riadku neprijme. Aby sa tak nestalo, boli využité funkcie `try` a `catch`, ktoré v prípade chyby neuložia žiadne znaky. Keby niektorý reťazec neobsahoval riadiaci znak, vypíše sa bez akejkoľvek úpravy. Na správne ukončenie komunikácie sa musí použiť funkcia `BeginInvoke`, ktorou sa dá predísť zablokovaniu užívateľského rozhrania, ako v nasledujúcej ukážke:

```
if (serialPort1.IsOpen)
    BeginInvoke(new EventHandler(delegate { serialPort1.Close(); }));
```

Odosielanie hodnôt do mikrokontroléru nie je okamžité, ale prebieha pomocou časovača `timerTransmitDelay`. Ten v každom prerušení poskladá reťazec s riadiacim symbolom a tromi číslami, ktorý následne odošle. Dvojciferné čísla sa doplnia o nulu na začiatku, aby sa zachoval pevný počet znakov, ako je znázornené tu:

```
string tx1 = textBoxServo1.Text.PadLeft(transmitValueLength, '0');
string tx2 = textBoxServo2.Text.PadLeft(transmitValueLength, '0');
string tx3 = textBoxServo3.Text.PadLeft(transmitValueLength, '0');
serialPort1.Write('@' + tx1 + tx2 + tx3);
```

Periód a týchto prerušení sa nastavuje v kolónke `Settings`. Časovač, samozrejme, nie je neustále zapnutý. V samotnej obsluhu prerušenia sa vždy vypne, ale zmena hodnoty v ktoromkoľvek `TrackBar` ho znova aktivuje. Týmto jednoduchým spôsobom sa zaistí odoslanie iba jedného reťazca za určitú dobu a nie pri každej inkrementácii premenných.

## 4.3 Obráz z kamery

Kvôli zjednodušeniu práce a štruktúry kódu v tomto projekte použili overené a funkčné DLL knižnice `AForge.Video.DirectShow.dll` a `AForge.Video.dll` dostupné z [21]. Každá nová snímka pripravená na prevzatie z kamery vyvolá prerušenie, v ktorom sa najprv vytvorí jej hlboká kópia a uloží do premennej typu `Bitmap`. Priame zapísanie snímky môže vyvolať problémy s pretekaním pamäte, kde sa kopírujú len referencie na originálnu snímku, ktorej sa systém nedokáže zbaviť, kým tieto referencie existujú. Ďalej je absolútne nevyhnutné, aby bola táto kópia vždy riadne vymazaná. Obsluha tohto prerušenia je znázornená tu:

```
void videoSource_NewFrame(object sender, NewFrameEventArgs eventArgs)
{
    if (camImage != null) camImage.Dispose();
    camImage = new Bitmap(eventArgs.Frame);
    pictureBoxWebCamVideo.Image = camImage;
}
```

## ZÁVER

Hlavný cieľ tejto práce, čiže návrh polohovacie zariadenie pre malú kameru, bol úspešne splnený. Bol vytvorený virtuálny model mechanizmu s výkresmi na výrobu všetkých jeho súčastí. Svojou hmotnosťou, veľkosťou a hranatým tvarom by sa pravdepodobne nehodil na použitie v malých lietajúcich strojoch, ale mohol by byť ideálnym príslušenstvom na väčšie pohyblivé časti robotov a strojov, kde je potrebné kameru manuálne natočiť aj na tretej osi rotácie. Ďalej môže slúžiť ako základ nového pokročilejšieho zariadenia s rôznymi senzormi na sledovanie polohy predmetu. Pridanie laserov alebo svetlometov namiesto kamery by malo úplne iný účel, ktorý môže byť v určitých prípadoch tiež veľmi užitočný.

Pri návrhu elektroniky sa zvolilo napájanie mikrokontroléru z počítača cez USB port, ktorý by priamo slúžil aj na komunikáciu. Servomotory sa ale musia napájať z iného zdroja. Na zistenie stavu napätia v týchto dvoch častiach obvodu sa použili LED diódy. Keby nastala situácia, v ktorej mikrokontrolér je vypnutý, ale pohony sú pod prúdom, PWM signál sa nevytvára a motory sa natočia do svojich limitných polôh. Aby sa tomu problému predišlo, druhá časť obvodu sa spína výkonovým tranzistorom pripojeným na jeden z pinov čipu.

Komunikácia pomocou UARTu nebola ošetrovaná na všetky možné poruchy. Chýbajúce znaky z reťazca sa nahrádzajú nulou a nič viac sa s tým nestane. Bolo by vhodné v tomto prípade zapísať do premenných ich pôvodné hodnoty alebo ich vôbec nemeniť. Ďalšia situácia, síce veľmi nepravdepodobná, môže nastať pri konverzii znakov reťazca, v ktorom sú aj písmená namiesto požadovaných čísel. Samozrejme, je možné tieto ochranné prvky doplniť, ale programy boli navrhnuté tak, aby k týmto problémom pri bežnej prevádzke nedochádzalo. Navyše niektoré funkcie priamo v sebe kontrolujú rozsahy čísel. V oboch vyššie zmienených prípadoch sa premenné nastaví na základnú hodnotu.

Obslužný softvér umožňuje užívateľovi zmeniť vlastnosti viacerých prvkov podľa potreby, napríklad rýchlosť inkrementácie polohy, hraničné hodnoty jednotlivých premenných a perióda odosielania údajov. Taktiež disponuje funkciou na automatické obnovenie komunikácie v prípade jeho nečakaného prerušenia. Z neznámych dôvodov program občas na pár sekúnd zamrzne pri takomto obnovení spojenia, čo je ale pri vhodnej manipulácii s vodičmi úplne zanedbateľné. Táto aplikácia umožňuje prehrávať video aj z iných zdrojov, nie len z použitej kamery, ale na komunikáciu pomocou UARTu so zariadeniami bez vhodného programu sa neodporúča.

Mechanizmus a doska s elektronikou boli v dobe písania tejto práce vo výrobe, takže nebolo možné overiť, či hotové zariadenie funguje podľa predpokladov.

## ZOZNAM POUŽITÝCH ZDROJOV

- [1] Rozdelenie a druhy bezpečnostných kamier CCTV. *OXE* [online]. [cit. 2015-04-27]. Dostupné z: <<http://www.oxe.sk/rozdelenie-a-druhy-bezpecnostnych-kamier-cctv/>>.
- [2] Commercial CCTV Security Cameras. *Samsung CCTV* [online]. [cit. 2015-04-28]. Dostupné z: <<https://www.samsung-security.com/products/security-cameras.aspx>>.
- [3] CMUcam: Open Source Programmable Embedded Color Vision Sensors. [online]. [cit. 2015-04-29]. Dostupné z: <<http://www.cmucam.org/>>.
- [4] Pan & Tilt Kit for Pi2Go-Lite and Pi Camera. *4tronix* [online]. [cit. 2015-04-29]. Dostupné z: <<http://4tronix.co.uk/blog/?p=527>>.
- [5] New Fatshark Pan/Tilt/Roll mount test. *FPVLAB* [online]. [cit. 2015-04-30]. Dostupné z: <<http://fpvlab.com/forums/showthread.php?22063-New-Fatshark-Pan-Tilt-Roll-mount-test>>.
- [6] Pan and Tilt Bases. *Endurance R/C* [online]. [cit. 2015-04-29]. Dostupné z: <<http://www.endurance-rc.com/pantilt.php>>.
- [7] What is a servo? *Servocity* [online]. [cit. 2015-04-23]. Dostupné z: <[https://www.servocity.com/html/what\\_is\\_a\\_servo\\_.html#.VTivvyGqpBe](https://www.servocity.com/html/what_is_a_servo_.html#.VTivvyGqpBe)>.
- [8] How do servos work? *Servocity* [online]. [cit. 2015-04-23]. Dostupné z: <[https://www.servocity.com/html/how\\_do\\_servos\\_work\\_.html#.VTOMYiGqpBd](https://www.servocity.com/html/how_do_servos_work_.html#.VTOMYiGqpBd)>.
- [9] Servo Motors – Information, Usage and Control. *Lirtex* [online]. [cit. 2015-04-23]. Dostupné z: <<http://www.lirtex.com/robotics/servo-motors-information-and-control>>.
- [10] Convert a servo into a high-torque motor. *Trevor's Bike Shed* [online]. [cit. 2015-05-20]. Dostupné z: <<http://vocaro.com/trevor/blog/2007/12/29/convert-a-servo-into-a-high-torque-motor/>>.
- [11] STEM Lesson: Hacking Hobby Servos For Continuous Rotation. *Teach STEM Now* [online]. [cit. 2015-04-23]. Dostupné z: <<http://www.teachstemnow.com/hacking-hobby-servos/>>.
- [12] Product. *Reflowster* [online]. [cit. 2015-04-23]. Dostupné z: <<http://reflowster.com/product.html>>.
- [13] Microcontroller. *Wikipedia, the free encyclopedia* [online]. [cit. 2015-04-23]. Dostupné z: <<http://en.wikipedia.org/wiki/Microcontroller>>.
- [14] Mikrokontroléry. *src.athaj.cz* [online]. [cit. 2015-04-23]. Dostupné z: <[http://src.athaj.cz/teaching/rev/mcu\\_intro](http://src.athaj.cz/teaching/rev/mcu_intro)>.

- [15] GRIDLING, G. - WEISS, B. Introduction to Microcontrollers. *Vienna Institute of Technology* [online]. 2007. Dostupné z: <<http://hopfen.topfen.net/uni/mc/MC-Things/Microcontroller.pdf>>.
- [16] Introduction to Microcontrollers. *Mikroelektronika* [online]. [cit. 2015-04-23]. Dostupné z: <<http://www.mikroe.com/chapters/view/64/chapter-1-introduction-to-microcontrollers/>>.
- [17] Pulse-width modulation. *Wikipedia, the free encyclopedia* [online]. [cit. 2015-04-23]. Dostupné z: <[http://en.wikipedia.org/wiki/Pulse-width\\_modulation](http://en.wikipedia.org/wiki/Pulse-width_modulation)>.
- [18] Arduino Serial, UART. *GitHub* [online]. [cit. 2015-05-04]. Dostupné z: <<https://github.com/CourseReps/ECEN489-Fall2014/wiki/Arduino-Serial,-UART>>.
- [19] Universal asynchronous receiver/transmitter. *Wikipedia, the free encyclopedia* [online]. [cit. 2015-04-23]. Dostupné z: <[http://en.wikipedia.org/wiki/Universal\\_asynchronous\\_receiver/transmitter](http://en.wikipedia.org/wiki/Universal_asynchronous_receiver/transmitter)>.
- [20] PIC16F887. *8-bit PIC® Microcontrollers* [online]. [cit. 2015-05-21]. Dostupné z: <<http://www.microchip.com/wwwproducts/Devices.aspx?product=PIC16F887>>.
- [21] AForge.NET. *Framework* [online]. [cit. 2015-05-13]. Dostupné z: <<http://www.aforgenet.com/aforge/framework/>>.

## ZOZNAM POUŽITÝCH SKRATIEK

<b>Skratka</b>	<b>Popis</b>
A/D, D/A	Analog-To-Digital, Digital-To-Analog
ADC	Analog-To-Digital Converter
ASCII	American Standard Code For Information Interchange
CCP	Capture/Compare/PWM
DC	Direct Current
DLL	Dynamic-Link Library
ECCP	Enhanced Capture/Compare/PWM
EEPROM	Electrically Erasable Programmable Read-Only Memory
EUSART	Enhanced Universal Synchronous/Asynchronous Receiver/Transmitter
HD	High-Definition
I2C	Inter-Integrated Circuit
IF	Interrupt Flag
ISR	Interrupt Service Routine
MCU	Micro Controller Unit
MSSP	Master Synchronous Serial Port
PTR	Pan-Tilt-Roll
PTZ	Pan-Tilt-Zoom
PWM	Pulse-Width Modulation
RAM	Random-Access Memory
ROM	Read-Only Memory
SFR	Special Function Register
SPI	Serial Peripheral Interface
SYN	Synchronous Idle
UART	Universal Asynchronous Receiver/Transmitter
USART	Universal Synchronous/Asynchronous Receiver/Transmitter
USB	Universal Serial Bus
WPF	Windows Presentation Foundation

## ZOZNAM OBRÁZKOV A TABULIEK

Obr. 1.1: Typy kamier [2]. a) dome kamera, b) otočná kamera .....	11
Obr. 1.2: CMUcam kamera [3]. .....	12
Obr. 1.3: Mechanizmus z plechov [4]. .....	12
Obr. 1.4: Držiak kamery Fat Shark [5]. .....	13
Obr. 1.5: Model mechanizmu PT-3 [6]. .....	13
Obr. 1.6: Komponenty servomotora [8]. .....	14
Obr. 1.7: Natočenie motora v závislosti na šírke pulzu [9]. .....	14
Obr. 1.8: Výbežok na zabránenie pretočenia [11]. .....	15
Obr. 1.9: Rôzne veľkosti mikrokontrolérov [12]. .....	15
Obr. 1.10: Schéma časovača a prerušenia [16]. .....	16
Obr. 1.11: PWM diagram. ....	17
Obr. 1.12: Prenos jedného znaku [18]. .....	18
Obr. 2.1: Osi rotácie kamery. ....	19
Obr. 2.2: Model mechanizmu. ....	20
Obr. 3.1: Kontaktné pole s elektronikou. ....	22
Obr. 3.2: Vývojový diagram obsluhy prerušenia. ....	23
Obr. 3.3: Vývojový diagram funkcie na čítanie jedného znaku. ....	24
Obr. 4.1: Hlavné okno programu. ....	25
 Tab. 3.1: Parametre PIC16F887 [20]. ....	 21

## **ZOZNAM PRÍLOH**

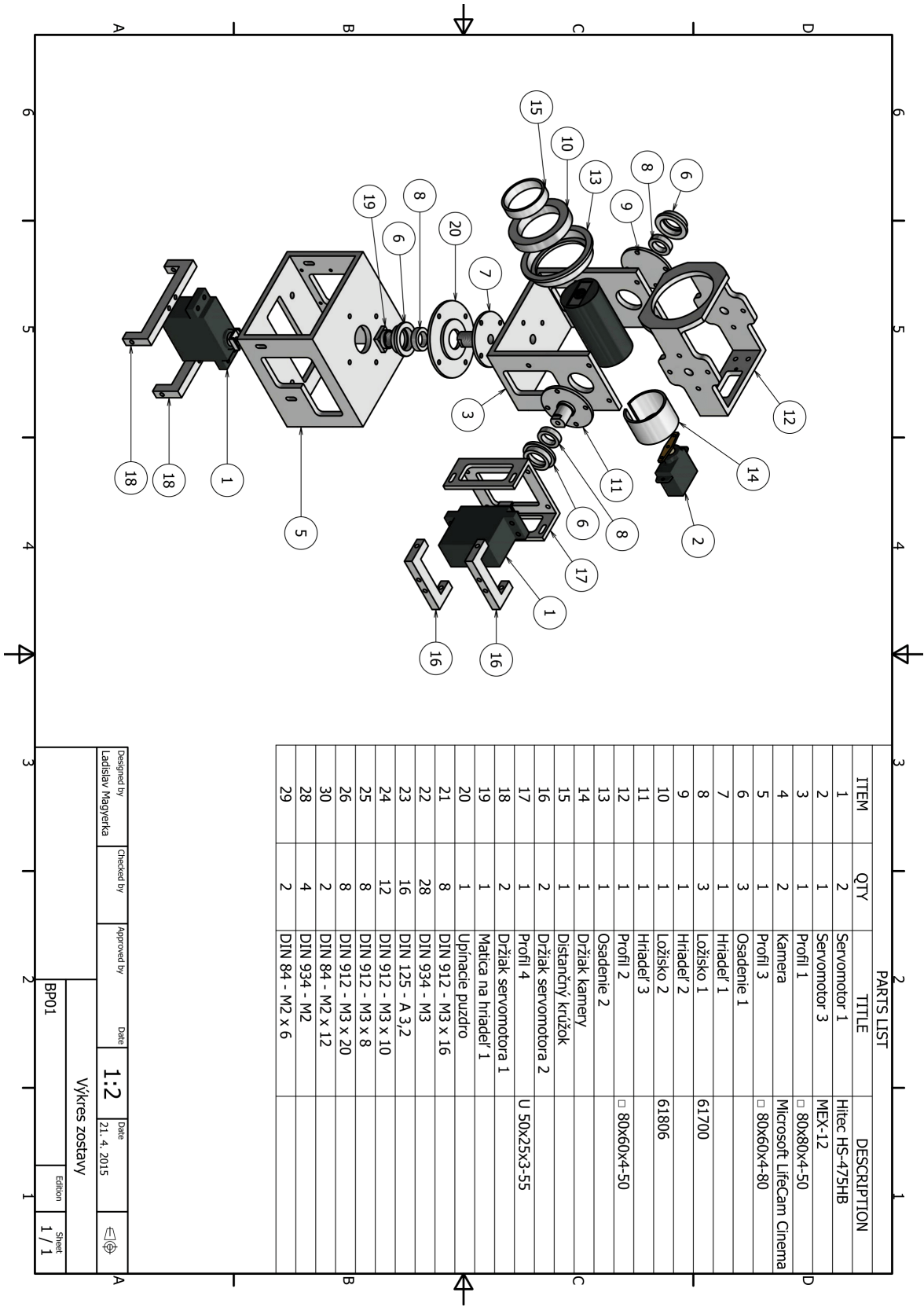
Príloha č.1 - CD s výkresom zostavy, schémou zapojenia, doskou plošných spojov a .exe súbor obslužného programu s potrebnými knižnicami

Príloha č.2 - Výkres zostavy mechanizmu

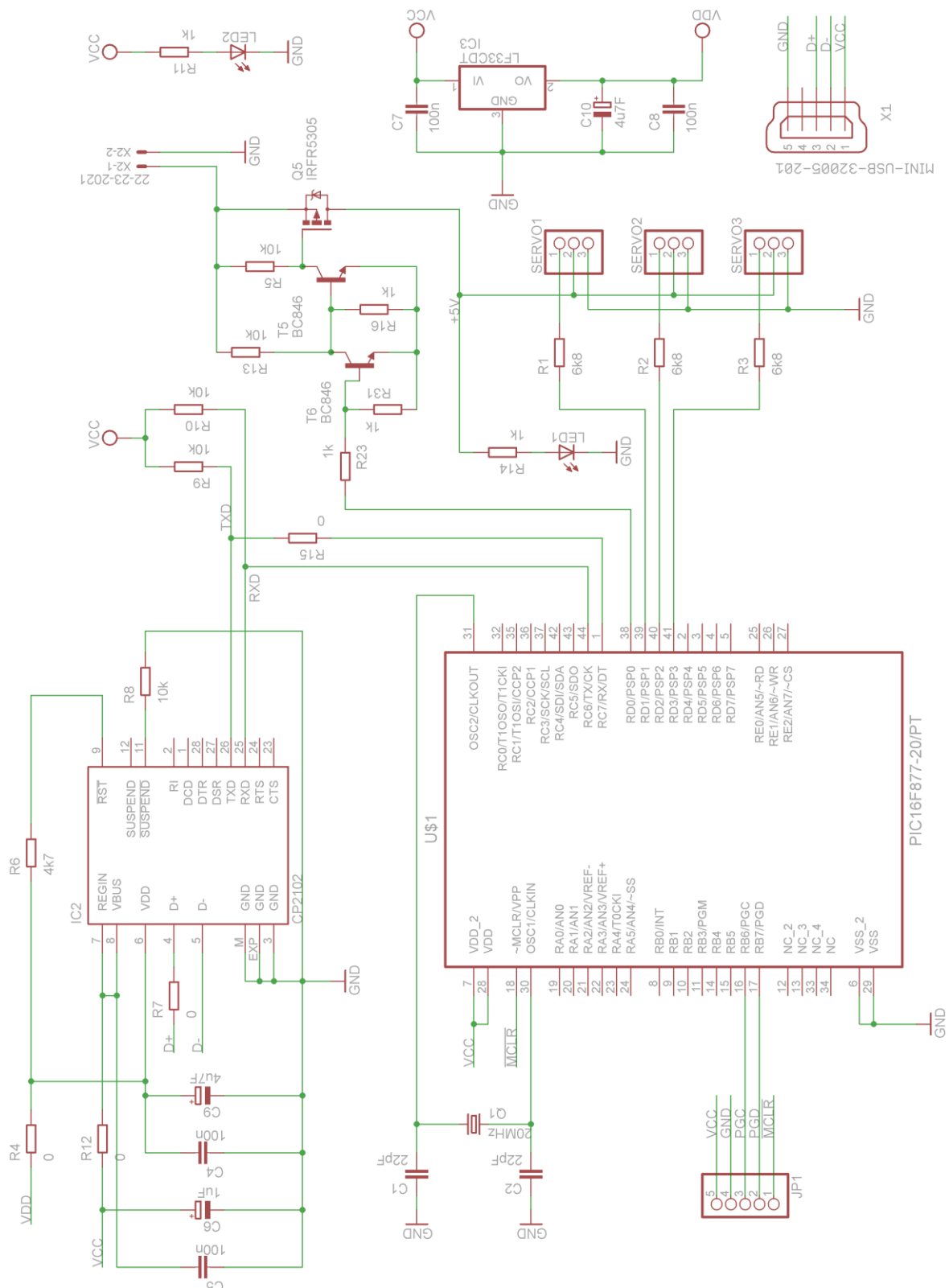
Príloha č.3 - Schéma zapojenia elektroniky

Príloha č.4 - Návrh dosky plošných spojov

PRÍLOHA Č.2 - VÝKRES ZOSTAVY MECHANIZMU



### PRÍLOHA Č.3 - SCHÉMA ZAPOJENIA ELEKTRONIKY



## PRÍLOHA Č.4 - NÁVRH DOSKY PLOŠNÝCH SPOJOV

