ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA STAVEBNÍ

STUDIJNÍ PROGRAM GEODÉZIE A KARTOGRAFIE

STUDIJNÍ OBOR GEODÉZIE A KARTOGRAFIE

DIPLOMOVÁ PRÁCE

AUTOMATICKÁ DETEKCE LOPATY MINIBAGRU V RÁMCI VZTAŽNÉ SOUSTAVY

Vedoucí práce: Ing. Zdeněk Vyskočil, Ph.D.

Katedra Geomatiky

Praha, 2019

Marousek Václav

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
Fakulta stavební
Thákurova 7, 166 29 Praha 6

ZADÁNÍ DIPLOMOVÉ PRÁCE

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: Marousek Jméno: Václav Osobní číslo: 439244

Zadávající katedra: K155 Katedra Geomatiky

Studijní program: (N3645) Geodézie a kartografie

Studijní obor: (3646T003) Geodézie a kartografie

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce: Automatická detekce lopaty minibagru v rámci vztažné soustavy

Název diplomové práce anglicky: Automatic detection of the minibagger bucket within the reference system

Pokyny pro vypracování:
Volba snímačů a testování
Konstrukce modelu
Tvorba aplikace pro Smartphone
Osazení systému na minibagru
Testování přesnosti

Seznam doporučené literatury:

Jméno vedoucího diplomové práce: Ing. Zdeněk Vyskočil, Ph.D.

Datum zadání diplomové práce: 22. 2. 2019 Termín odevzdání diplomové práce: 19.5.2019
Údaj uveďte v souladu s datem v časovém plánu příslušného ak. roku

[redacted]
Podpis vedoucího práce

[redacted]
Podpis vedoucího katedry

III. PŘEVZETÍ ZADÁNÍ

Beru na vědomí, že jsem povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je nutné uvést v diplomové práci a při citování postupovat v souladu s metodickou příručkou ČVUT „Jak psát vysokoškolské závěrečné práce" a metodickým pokynem ČVUT „O dodržování etických principů při přípravě vysokoškolských závěrečných prací".

22. 2. 2019
Datum převzetí zadání

[redacted]
Podpis studenta(ky)

Prohlášení

Prohlašuji, že diplomovou práci na téma *Automatická detekce lopaty minibagru v rámci vztažné soustavy* jsem vypracoval samostatně pod vedením Ing. Zdeňka Vyskočila, Ph.D.. Dále jsem využil zdroje, které uvádím v seznamu literatury.

V Praze dne …… Podpis ……..

Poděkování

Touto cestou bych chtěl poděkovat vedoucímu práce Ing. Zdeňku Vyskočilovi, Ph.D za cenné rady během zpracování diplomové práce a také Jiřímu Emlerovi za poskytnutí minibagru pro realizaci experimentu. Dále bych chtěl poděkovat především své rodině a přítelkyni za neustálou podporu při studiu a při tvorbě diplomové práce.

Abstrakt

Tato diplomová práce se zabývá vývojem zařízení pro detekci lopaty minibagru v rámci vztažné soustavy pomocí snímačů pro měření vzdálenosti a řídícího zařízení *Arduino*. Nejprve bylo provedeno testování různých snímačů na délkové základně v laboratoři *FSV ČVUT*, následně byly dle přesnosti a pracovního rozsahu zvoleny snímače pro následnou konstrukci měřické soustavy. Pro čtení dat z měřické soustavy byla vytvořena aplikace pro platformu Android ve vývojovém prostředí Android Studio ve skriptovacím jazyce Java. Celý systém byl testován na modelu a také na skutečném minibagru.

Klíčová slova

Snímač, Arduino, Měřící systém, Aplikace, Android studio

Abstract

This diploma thesis deals with the development of a device for detecting bucket of a mini excavator within the reference system by means of sensors for distance measurement and control device Arduino. First, various sensors were tested at the length base in the FSV CTU laboratory, and sensors for the subsequent construction of the measuring system were selected according to the accuracy and working range. An application for the android platform was developed for reading data from the survey system in the development environment Android Studio in language Java. The whole system was tested on the model and also on the real minibagger.

Keywords

Sensor, Arduino, Measuring System, Application, Android Studio

Obsah

Úvod ..... - 8 -
1. Volba snímačů a testování ..... - 9 -
1.1. Laserový snímač vzdálenosti "GY-VL53L0X" ..... - 9 -
1.1.1. Testování - měření jedním snímačem ..... - 10 -
1.1.2. Testování - měření třemi snímači současně ..... - 11 -
1.1.3. Výsledky testování ..... - 12 -
1.2 Ultrazvukový snímač vzdálenosti "HC-SR04" ..... - 13 -
1.2.1. Testování - Fáze 1 ..... - 14 -
1.2.2. Testování - Fáze 2 ..... - 15 -
1.2.3. Výsledky testování ..... - 16 -
1.3. Snímač teploty a vlhkosti vzduchu "DHT11" ..... - 17 -
1.4. Arduino Mega 2560 ..... - 17 -
1.5. Použité totální stanice ..... - 18 -
1.5.1. Trimble S7 robotic ..... - 19 -
1.5.2. Leica TC1700 ..... - 19 -
1.5.3. Leica TC307 ..... - 20 -
1.6. Laserový snímač vzdálenosti HI50 ..... - 21 -
1.7. Bluetooth modul HC-05 ..... - 22 -
2. Konstrukce modelu ..... - 23 -
2.1. Stavba modelu ..... - 23 -
2.2. Výpočet chybového modelu ..... - 24 -
2.3. Testování ..... - 25 -
3. Tvorba aplikace pro Smartphone ..... - 27 -
3.1. Volba cílové platformy ..... - 27 -
3.2. Základní dělení aplikací dle způsobu vývoje ..... - 27 -
3.2. MIT Inventor 2 ..... - 29 -
3.3. Android Studio ..... - 31 -
3.4. Aplikace "Bagr" ..... - 32 -
3.4.1. Úvodní obrazovka ..... - 32 -
3.4.2. Hlavní obrazovka ..... - 33 -
3.5. Výpočet ..... - 35 -
4. Osazení systému na minibagru ..... - 39 -
4.1. Kubota KX 71-3 ..... - 39 -
4.2. Testování ..... - 39 -
5. Závěr ..... - 41 -

6. Seznam zdrojů ..... - 43 -
7. Seznam obrázků ..... - 46 -
8. Seznam tabulek ..... - 47 -
9. Seznam schémat zapojení ..... - 48 -
10. Seznam zkratek ..... - 49 -
11. Seznam příloh ..... - 50 -

Úvod

Cílem této diplomové práce je konstrukce měřící soustavy pro detekci lopaty minibagru s co nejmenšími pořizovacími náklady a dostačující přesností.

Měřící soustava je složena ze snímačů, které jsou řízeny pomocí vývojové desky *Arduino*, na kterou je navíc připojen *Bluetooth* modul, který zajišťuje přenos naměřených veličin do mobilního zařízení. Pro zpracování těchto měřených veličin, byla ve vývojovém prostředí *Android studio* od společnosti *Google*, vytvořena aplikace s názvem "Bagr". Aplikace uživateli nabízí využití měřické soustavy nejen na minibagru Kubota KX71-3, ale i na dalších strojích, a to především díky snadnému vložení pevných rozměrů ramene. Aby uživatel věděl, které rozměry je nutno vložit, obsahuje aplikace také náčrt těchto parametrů včetně označení. Výpočet 2D souřadnic (ve svislé rovině procházející ramenem bagru) je založen na protínání z délek a sinové resp. kosinové větě, pomocí kterých jsou vypočítány úhly. Souřadnice byly vypočteny volným polygonovým pořadem.

Model ramene, byl vytvořen pro testování funkčnosti měřické soustavy a také správnost výpočtu souřadnic, který probíhá uvnitř mobilní aplikace. Vypočtené výškové rozdíly byly porovnány s výškovými rozdíly naměřených totální stanicí Trimble S7 robotic.

Testování měřické soustavy bylo provedeno také na minibagru Kubota KX71-3 a výsledné výškové rozdíly zobrazené v aplikaci byly porovnány s výškovými rozdíly určenými totální stanicí Leica TC1700.

Výsledkem této práce je měřická soustava, kterou je možné ovládat pomocí původní aplikace v chytrém telefonu s operačním systémem Android.

1. Volba snímačů a testování

V této kapitole popíšu snímače, které byly použity v rámci vývoje měřické soustavy. Dále jsou zde popsány postupy testování a parametry použitých totálních stanic.

1.1. Laserový snímač vzdálenosti "GY-VL53L0X"

Snímač "GY-VL53L0X" je laserový snímač vzdálenosti, který funguje na principu měření tranzitního času od vyslání do příjmu světelného paprsku (IR laserová stopa s vlnovou délkou 940nm). Dle dokumentace od výrobce má zařízení přesnost ±3 % a rozlišovací schopnost jednotky milimetrů. Pracovní rozsah výrobce udává max. 2 metry.

Obr. 1 Laserový snímač vzdálenosti "GY-VL53L0X" [19]

Snímač umožňuje měřit ve čtyřech režimech (1. Výchozí režim, 2. Vysoká rychlost, 3. Vysoká přesnost, 4. Dlouhý dosah). Snímač komunikuje s *Arduinem* pomocí digitálních pinů "SDA" a "SCL" (I2C komunikace), dále pro definici adresy slouží pin "X-SHUT" a pro přerušení měření pin "GP|01". Zařízení je napájeno napětím 5V.

[1]

1.1.1. Testování - měření jedním snímačem

První test laserového snímače byl proveden na délkové základně v laboratoři FSV ČVUT v Praze při teplotě *20 °C* (teplota při které bylo zařízení testováno výrobcem). Tento snímač je ovládán pomocí knihovny *VL53l0X.h*, která byla stažena z internetu. Laserový snímač byl zapojen na *Arduino* dle následujícího schématu.

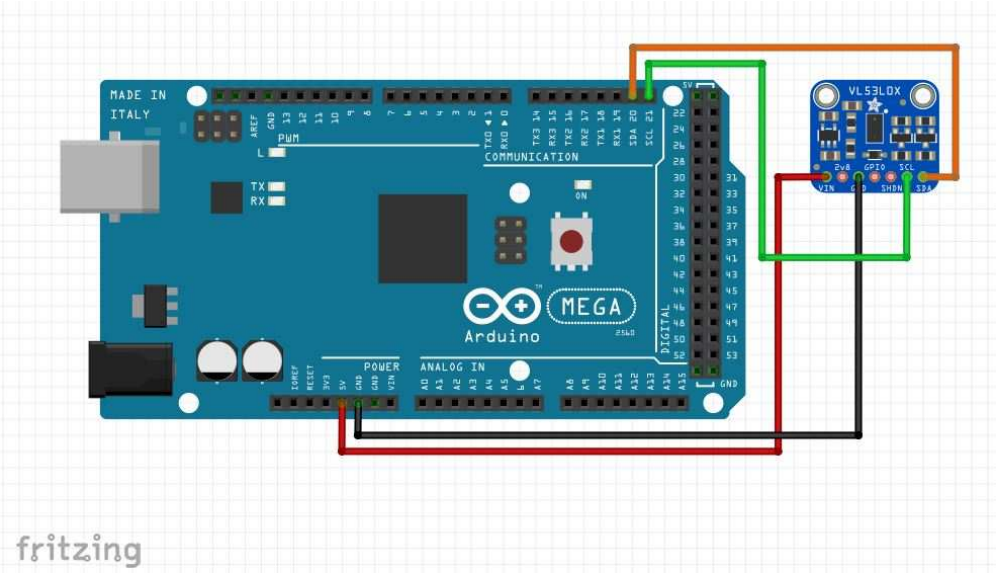


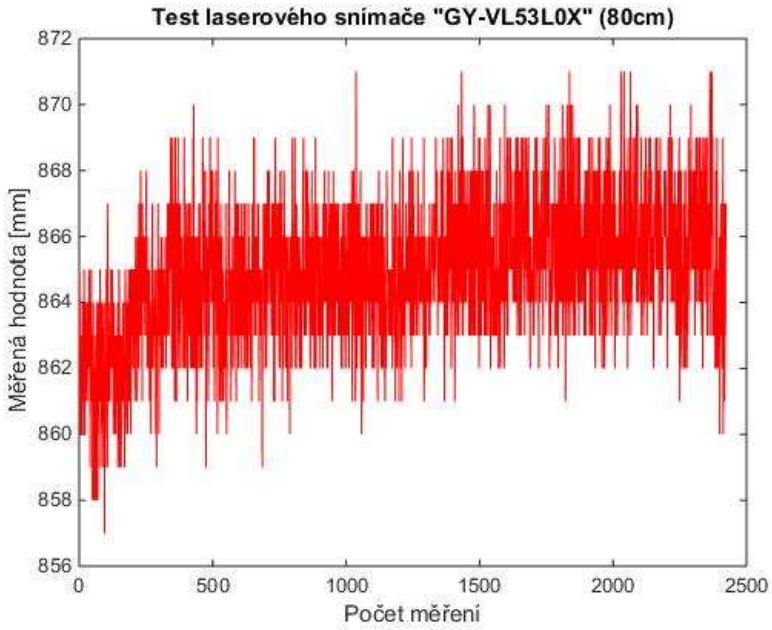
Schéma 1.GY- VL53L0X

Pomocí pinu *X-SHUT* je nastavována adresa snímače. Pro měření pouze jedním snímačem není nutné tento pin používat.

Pojezdový vozík základny byl osazen cílem z polystyrenu o rozměru (*33,5 x 20,5 cm*) a od snímače byl vzdálený *80 cm*. Snímač byl spuštěn v režimu "dlouhý dosah". Pro určení jedné vzdálenosti byl vypočten aritmetický průměr z 10 naměřených hodnot. Smyčka probíhala s opakováním 500 milisekund po dobu 15 minut. Naměřená data byla následně importována do Matlabu, kde proběhl výpočet směrodatné odchylky výběru. Výsledkem je graf naměřených hodnot a směrodatná odchylka σ výběru pro vzdálenost *80 cm*.

$$\sigma = 2.1\ mm$$

[2] [22]



Obr. 2 Graf měřených hodnot "GY-VL53L0X"

1.1.2. Testování - měření třemi snímači současně

Pro testování více snímačů najednou je nutné použití pinu X-SHUT a následná definice adres jednotlivých laserových snímačů. Zařízení byla k *Arduinu* připojena dle následujícího schématu.

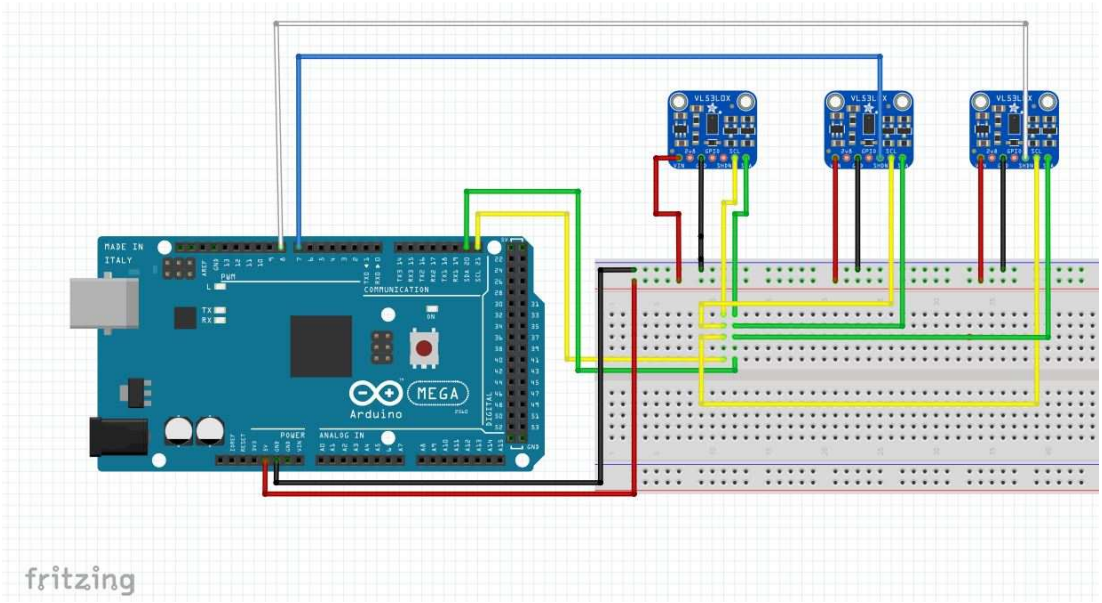


Schéma 2. Test tří snímačů GY-VL53L0X současně

Při tomto testování byly proměřeny vzdálenosti v rozsahu od *20 cm* do *120 cm* (po *10 cm*), přičemž *120 cm* v laboratorních podmínkách se ukázalo jako

maximální měřitelná vzdálenost při aktuální velikosti cíle (*33,5 x 20,5 cm*). Z toho vyplývá, že hodnota udávaná výrobcem (200cm) není reálná a pro její dosažení by musel být pravděpodobně zvětšen cíl, jelikož laserový svazek není usměrněný a má tvar kužele.

Každá vzdálenost byla měřena 3 minuty s opakováním po 0,5 sekundy. Měřená data byla načtena do Matlabu a proběhl výpočet výběrových směrodatných odchylek, které jsou sestaveny do tabulky (*Tab 1.*). Měřená data byla zakreslena do grafů a jsou součástí přílohy.

[2]

1.1.3. Výsledky testování

Z výsledků vypočtených z měřených hodnot je patrné, že přesnost laserového snímače klesá společně s rostoucí měřenou vzdáleností. I přes tuto skutečnost však tento snímač dosahuje vysoké přesnosti. Snímač je bohužel limitován neusměrněným paprskem, což způsobuje nutnost cíle velkých rozměrů a to pro náš systém není vhodné. Na základě testování jsem se proto rozhodl tento snímač již dále nepoužívat. Určitě bych ho ale vřele doporučil například na osazení robota pro detekci překážek a to především díky malým rozměrům snímače a vysoké přesnosti.

Testování tří snímačů GY-VL53L0X současně			
Snímač č. 1		**Snímač č. 2**	**Snímač č. 3**
Vzdálenost	**Směrodatná odchylka**	**Směrodatná odchylka**	**Směrodatná odchylka**
[m]	*[mm]*	*[mm]*	*[mm]*
0,2	2,2	3,8	3,7
0,3	2,5	3,7	3,7
0,4	3,2	3,9	4,0
0,5	4,2	8,6	8,2
0,6	5,2	8,2	7,6
0,7	6,3	7,8	7,8
0,8	8,6	8,3	7,5
0,9	9,6	9,1	8,6
1,0	11,8	9,8	10,0
1,1	14,3	10,7	11,1
1,2	16,4	10,7	11,9

Tab 1. Test tří snímačů GY-VL53L0X současně

1.2 Ultrazvukový snímač vzdálenosti "HC-SR04"

Snímač "HC-SR04" je ultrazvukový snímač pro měření vzdálenosti, který funguje na principu měření tranzitního času od vyslání do příjmu zvukového signálu. Snímač je osazený reproduktorem a mikrofonem, které jsou napájeny na desku vedle sebe. Tento modul umožňuje detekci vzdálenosti v rozsahu 2cm - 4m s rozlišením jednotek centimetrů. Důležitý je také pracovní úhel detekce α <15°. Zařízení je napájeno napětím 5V.

Obr. 3 Ultrazvukový snímač vzdálenosti "HC-SR04" [20]

Tranzitní čas je zaznamenávám v *cm/μs*, pro výpočet vzdálenosti musíme použít tabelovanou hodnotu rychlosti zvuku (dle teploty např. při teplotě $t_0 = 20° C$ je rychlost zvuku $v = 343m/s$). Výsledná vzdálenost je tedy vypočtena ze vzorce

$$s = \frac{v * t}{2}$$

kde *s* je vzdálenost *(cm)*, *v* je rychlost zvuku *(cm/μs)* a *t* je tranzitní čas *(μs)*.

Pro zachování přesnosti senzoru při různých teplotách byl do testu implementován teplotní senzor "*DHT11*" a pro výpočet vzdálenosti je použit cyklus, který vkládá konstantu rychlosti vzduchu dle teploty okolního prostředí.

[3] [4]

1.2.1. Testování - Fáze 1

Pří testování ultrazvukového dálkoměru byl vždy současně použit snímač teploty a vlhkosti vzduchu *DHT11* kvůli zpřesnění výsledných vzdáleností. Snímače byly zapojeny v následujícím schématu.

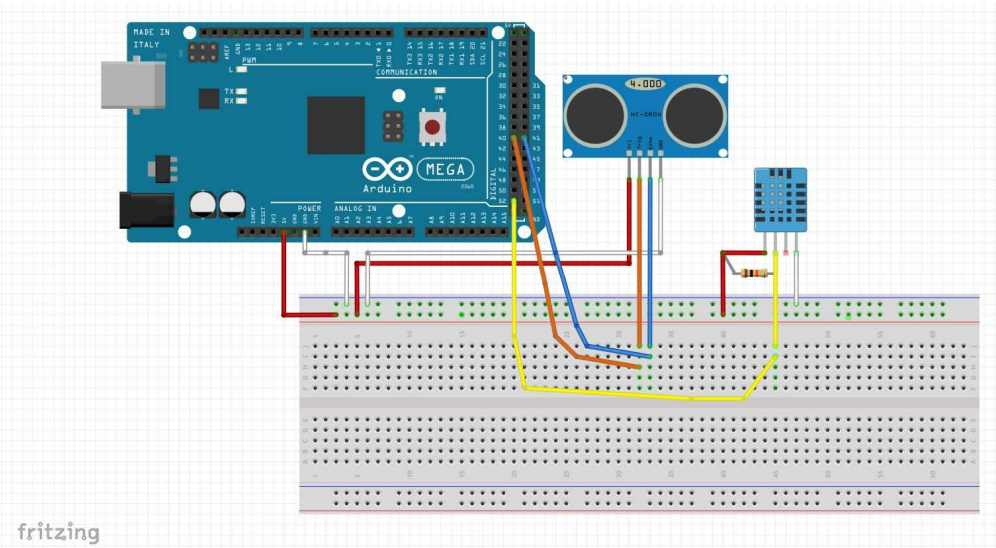


Schéma 3. HC-SR04 + DHT11

Hned při prvním spuštění měření se snímačem *HC-SR04* byly zpozorovány veliké výchylky mezi měřenými hodnotami. Ty jsou způsobovány kvůli několikanásobnému odrazu zvuku. S tímto snímačem se již od vzdálenosti 60 cm navíc nedá téměř zacílit. Při snaze tento problém vyřešit jsem nejprve na vysílač nasadil pět centimetrů dlouhou plastovou trubičku pro lepší usměrnění zvuku ale výsledek se nedostavil. Snímač stále měřil s vysokým rozptylem hodnot a proto jsem se rozhodl k následujícímu kroku. Ze základové desky snímače jsem pomocí pájky odstranil přijímač, který jsem k desce opětovně připojil pomocí drátů. Díky tomu, že nyní je umístěný přijímač přímo proti vysílači se velice snížil rozptyl měřených hodnot a navíc vznikla umělá signalizace přímo pomocí přijímače.

Jelikož se změnila konstrukce snímače, je také nutné upravit vzorec pro výpočet vzdálenosti

$$s = v * t$$

kde *s* je vzdálenost *(cm)*, *v* je rychlost zvuku *(cm/μs)* a *t* je tranzitní čas *(μs)*.

[3] [6]

1.2.2. Testování - Fáze 2

Abych prokázal výhodu úpravy ultrazvukového snímače, rozhodl jsem se provést testování, při kterém porovnám vzdálenosti měřené pomocí upraveného a neupraveného snímače. Pro měření více snímači najednou byla použita knihovna *"NewPing"*. K počítači byla tedy připojena totální stanice Leica TC307, laserové disto *HI-AT Technology CO.,LTD* a *Arduino* s připojenými snímači "*HC-SR04*".

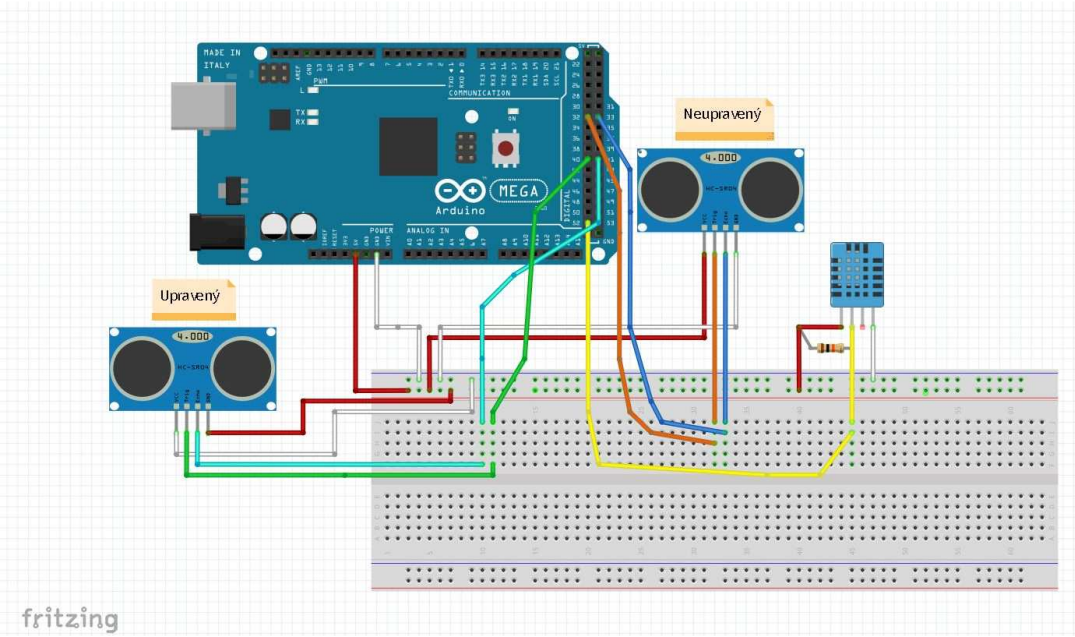


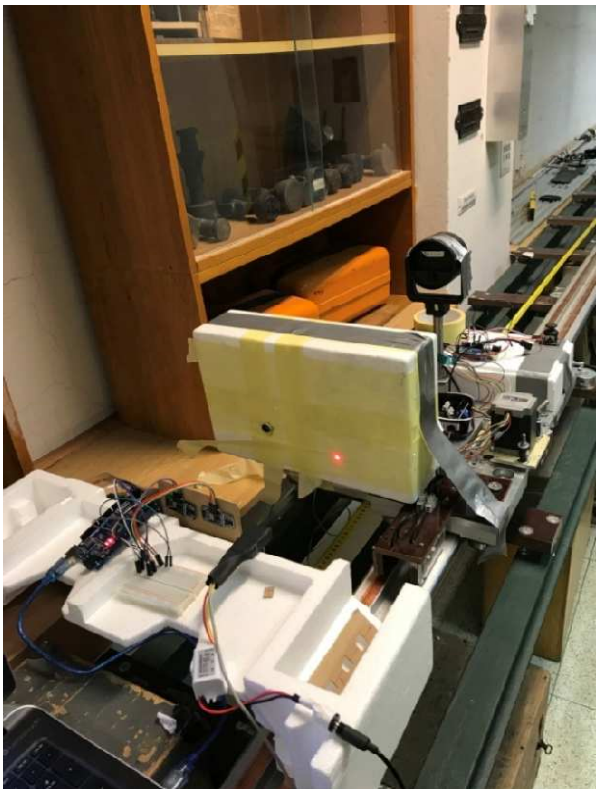
Schéma 4. 2 x HC-SR04 + DHT11

Na pojezdový vozík byl k cíli navíc umístěn ještě odrazný hranol, na který byla zacílena totální stanice. Cíl byl posouván přibližně po deseti centimetrech a v každém úseku byl proveden ještě malý posun (max *1 cm*) kvůli testování rozlišení snímačů. Zařízení byla ovládána v *Matlabu* přes sériovou komunikaci. Měření bylo provedeno ve čtyřech jednotkách (*1* jednotka = *1 x* měření TS + *5 x* měření snímači). Data byla registrována do textových souborů rozdělených podle proměřované vzdálenosti. Pro výpočet směrodatných odchylek bylo použito měření totální stanicí *Leica TC307*, které jsem považoval ve výpočtu za bezchybné, a měřená data ze snímačů. Výsledné směrodatné odchylky jednotlivých snímačů jsou uvedeny v tabulce (*Tab. 2*).

[5] [6]

1.2.3. Výsledky testování

Tímto testem bylo prokázáno, že úprava snímače zlepší jeho přesnost a je možné jeho použití pro osazení minibagru s možností kombinace s laserovým dálkoměrem HI50 pro zlepšení koncové přesnosti. Při testování bylo nutné podložení základny kde byly umístěny snímače. Při tom se uvolnil snímač *HI50* a z toho důvodu je vypočtená směrodatná odchylka výrazně horší oproti té udávané výrobcem. Jelikož jsem tento snímač nakonec nepoužil nebylo nutné toto testování opakovat. Pro volbu snímačů byl napsán skript v *Matlabu*, kde je vypočítána koncová přesnost polohy lopaty minibagru na základě přesnosti osazených snímačů.



Obr. 4 Testování snímačů - délková základna FSV ČVUT

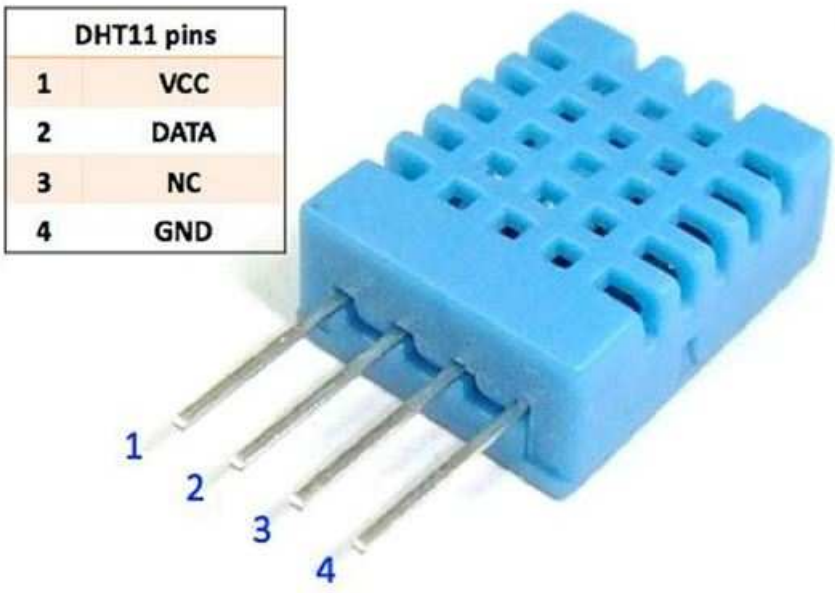
Testování ultrazvukových dálkoměrů - Fáze 2		
Ultrazvukový dálkoměr - Neupravený	**Ultrazvukový dálkoměr - Upravený**	**Laserový dálkoměr - HI50**
Sm. Odchylka	**Sm. Odchylka**	**Sm. Odchylka**
[m]	*[m]*	*[m]*
0,489	0,016	0,010

Tab 2. Test úpravy ultrazvukového dálkoměru

1.3. Snímač teploty a vlhkosti vzduchu "DHT11"

DHT11 je jednoduše použitelný snímač pro měření teploty a vlhkosti okolního prostředí. Díky jeho nízké ceně a dobré přesnosti měření je pro účel projektu vhodný.

K úspěšnému přenosu dat mezi snímačem a *Arduino* deskou je zapotřebí propojit tři vodiče, mezi VCC (5V) a digitálním datovým pinem je zapotřebí vložit rezistor *10kΩ* . Třetí vodič propojíme se zemí *Arduina* a můžeme začít číst data. Snímač využívá integrovanou knihovnu *DHT.h.*



Obr. 5 Snímač DHT11 [11]

1.4. Arduino Mega 2560

Arduino Mega 2560 je vývojová deska s mikrokontrolerem ATmega 2560. Oproti svému předchůdci *Arduino Uno* má *Arduino Mega 2560* větší rozměry a více vstupů a výstupů, přesně disponuje padesáti čtyřmi digitálními vstupy a výstupy, šestnácti analogovými vstupy a čtyřmi rozhraními UART. Připojení desky k počítači probíhá přes USB-B konektor, dále je na desce konektor pro připojení externího napájení, nemůže také chybět tlačítko reset, kterým jednoduše resetujeme desku resp.

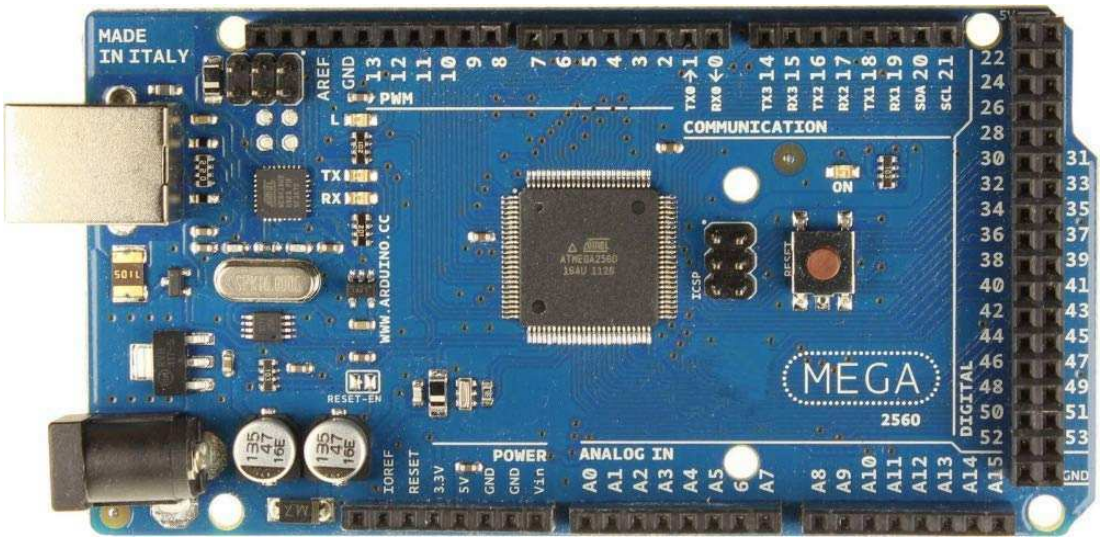
restartujeme nahraný skript. *Arduino Mega* má oproti předchůdci větší operační paměť a tím pádem i větší výpočetní výkon.

Limity vstupního napětí jsou od *6* do *20V*, výrobcem doporučené vstupní napětí je uváděno na *7-12V*.

Délka celého zařízení je *101,52* mm, šířka *53,3* mm a výška *15,1* mm, hmotnost 37 g. Taktovací frekvence desky je *16MHz*.

Arduino projekty jsou vyvíjeny v grafickém vývojovém prostředí, které obsahuje vzorové projekty a také nápovědu.

[10]



Obr. 6 Arduino Mega 2560 [12]

1.5. Použité totální stanice

Pro testování modulů a měřické soustavy byly použity celkem tři totální stanice. Tato obměna byla způsobena tím, že při každém testování byl dostupný jiný přístroj.

1.5.1. Trimble S7 robotic

Trimble S7 robotic je totální stanice, která uživateli dává spoustu možností. Totální stanice je ovládána pomocí servomotorů, které jsou všechny umístěny na jedné straně přístroje, což velice usnadňuje práci s přístrojem.

Datasheet:

Měření úhlů:
Přesnost (směrodatná odchylka podle DIN 18723).....2" (0.6 mgon)
Měření délek:
Přesnost (směrodatná odchylka)
Na hranol
Standard..... (2 mm + 2 ppm)
Tracking.....(4 mm + 2 ppm)

[13]

1.5.2. Leica TC1700

Leica TC1700 je totální stanice ze série TPS1000, která umožňuje registraci dat do paměti. Totální stanice je ovládána pomocí hrubých a jemných mechanických ustanovek.

Datasheet:
Měření úhlů:
Přesnost (směrodatná odchylka podle DIN 18723).....1.5" (0,45 mgon)
Měření délek:
Přesnost (směrodatná odchylka)
Na hranol
Standard..... (2 mm + 2 ppm)

[14]

1.5.3. Leica TC307

Leica TC307 je totální stanice, která umožňuje registraci dat do paměti (až 8000 bodů). Totální stanice je ovládána pomocí nekonečných šroubů a je bez mechanických ustanovek. Komunikace s totální stanicí byla provedena přes sériový port. Měřené vzdálenosti byly načítány pomocí příkazu "fprintf(port_3,'GET/M/WI31')" a následně byl obdržený kód převeden do standardního formátu.

Datasheet:

Měření úhlů:

Přesnost (směrodatná odchylka podle DIN 18723).....7" (2 mgon)

Měření délek:

Přesnost (směrodatná odchylka)

Na hranol

Standard..... (2 mm + 2 ppm)

[14]

1.6. Laserový snímač vzdálenosti HI50

HI50 od *HI-AT Technology CO.,LTD* je laserový dálkoměr, který je možné připojit k PC přes sériový port. Rozsah měření tohoto snímače je *0.02 - 50* metrů s udávanou přesností ± *1mm* na odraznou desku za ideálních světelných podmínek. Při špatných okolních podmínkách výrobce udává přesnost *3 mm + 40ppm*. Rozměry tohoto snímače jsou *45x25x12 mm*.

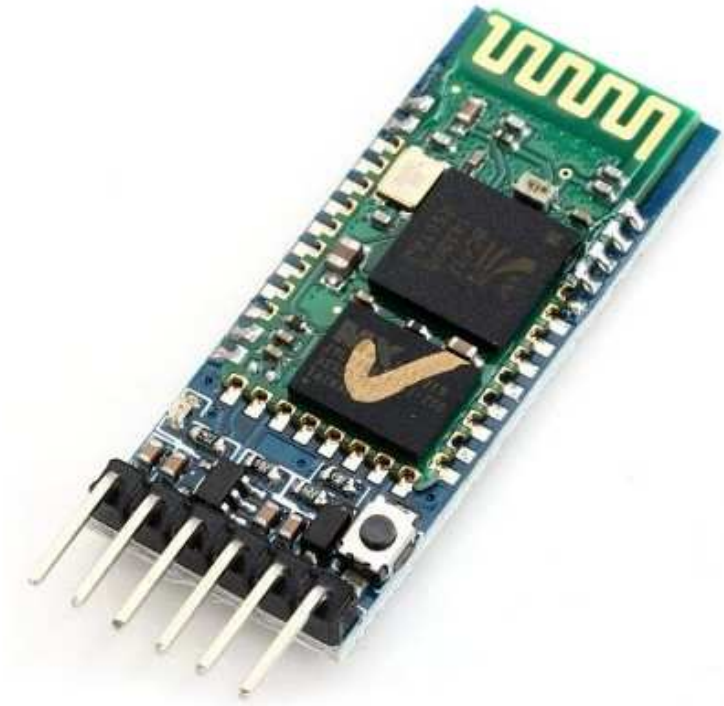
Obr. 7 Snímač vzdálenosti HI50 [21]

Tento snímač používá červený laserový paprsek o vlnové délce *635 nm*. Spadá do bezpečnostní třídy II. V této třídě jsou lasery s viditelnou stopou, které při chvilkovém ozáření jsou bezpečné. Mohou být však nebezpečné při přímém pohledu do laserového svazku. Proto je nutné s tímto snímačem zacházet opatrně a neohrozit tím nikoho v okolí.

[18] [24]

1.7. Bluetooth modul HC-05

Bluetooth modul HC-05 slouží k propojení *Arduina* s jiným zařízením podporujícím *Bluetooth* bez použití vodičů. Modul obsahuje Bluetooth verze 2.0. Komunikace s Arduinem je po sériové lince s defaultní přenosovou rychlostí *9600 baudů*. Anténa modulu je však poměrně malá, což způsobuje menší dosah přenosu (max. *10 m*). Napájecí napětí je v rozmězí *3,3* až *6 Voltů*. Rozměry modulu jsou pouhých *32 x 16 mm.*



Obr. 8 Bluetooth modul HC-05 [23]

Tento modul byl k *Arduinu* připojen pomocí čtyř vodičů, které jsou zakresleny ve *schéma č.5* (*Kapitola 2.1. Stavba modelu*).

[23]

2. Konstrukce modelu

V této kapitole popíšu stavbu modelu, výpočet chybového modelu a samotné testování měřícího systému na modelu.

2.1. Stavba modelu

Model ramene minibagru byl vytvořen v dílně ČVUT fakulty stavební. Pro simulaci pístů byly zvoleny "jekly" (duté železné hranoly). Na každý píst byly použity dva hranoly různé velikosti tak, aby bylo možné je do sebe zasunout. Konstrukce ramene byla vytvořena z dřevěných hranolů, které byly spojeny pomocí šroubů a matek. Na klouby modelu byly tavnou pistolí přilepeny polystyrenové nosiče na snímače, které byly následně osazeny samotnými snímači.

Snímače *HC-SR04* byly připojeny pomocí vodičů na jednu desku *Arduino Mega 2560* společně se snímačem *DHT11* a *Bluetooth* modulem *HC-05*. Zapojení bylo provedeno dle následujícího schéma.

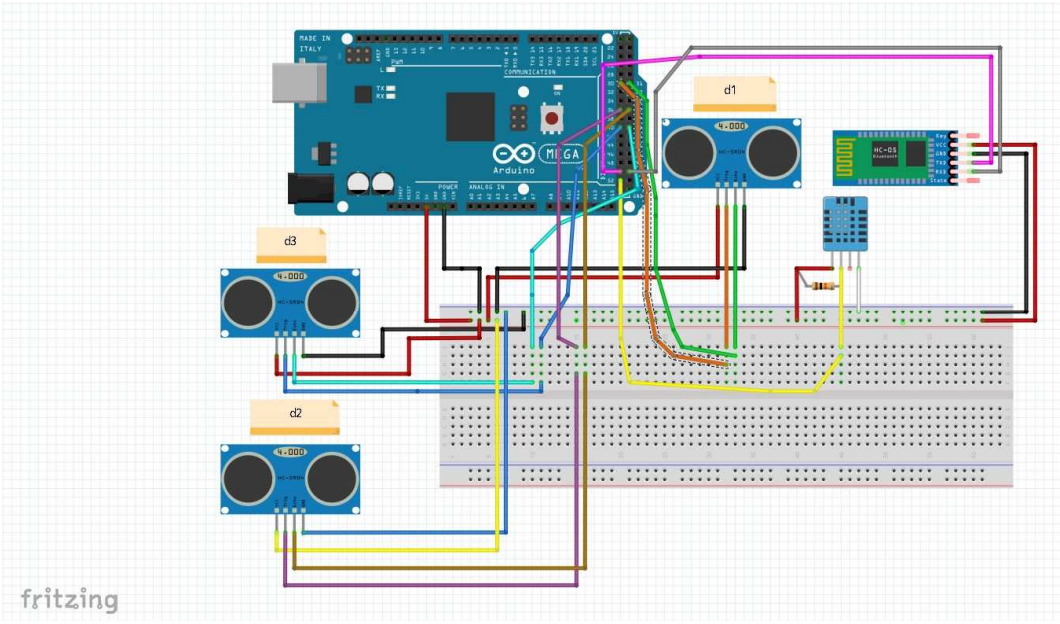
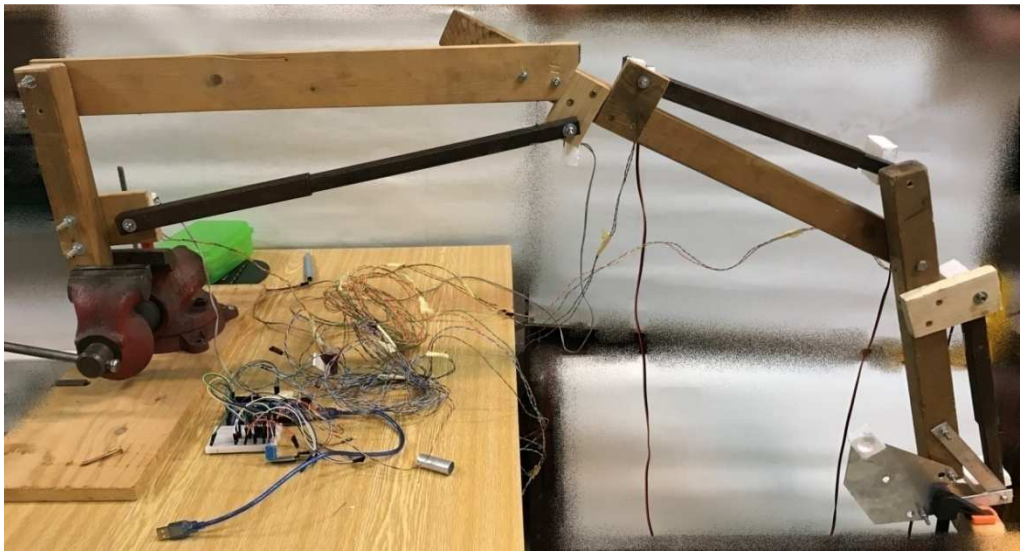


Schéma 5. Kompletní zapojení měřického systému

Model sloužil především kvůli vyladění chyb ve výpočtu a v aplikaci, dále také kvůli testování funkčnosti celé měřické soustavy.



Obr. 9 Model ramene minibagru

2.2. Výpočet chybového modelu

Přesnost určení souřadnic zubu lopaty byla vypočítána stejně jako přesnost koncového bodu určeného polygonovým pořadem. Výpočet byl proveden v *Matlabu*. Nejprve byl sestaven nelineární matematický model volného polygonového pořadu a poté byla sestavena matice **Q** neboli matice parciálních derivací podle všech měřených veličin. Tato matice má tedy rozměr *1 x 3* jelikož mám pouze 3 měřené veličiny.

Dále byla vypočtena kovarianční matice

$$\boldsymbol{\Sigma_x = Q\,P\,Q^T}\,.$$

Z kovarianční matice byla vypočtena směrodatná odchylka souřadnice *X* resp. *Z*, jelikož souřadná soustava byla zvolena v rovině YZ, jako odmocnina z diagonálních prvků. V mém případě byla kovarianční matice rozměru 1x1, protože řeším pouze výškové rozdíly

$$\sigma_x = \sqrt{(\boldsymbol{\Sigma_x})_{1,1}}\,.$$

Při použití tří upravených ultrazvukových dálkoměrů při středních rozměrech všech tří pístů jak modelu tak minibagru. Vychází směrodatná odchylka koncového bodu viz (*Tab.3*).

Přesnost koncového bodu	
Sm. Odch - model	**Sm. Odch - minibagr**
[mm]	*[mm]*
75,7	75,8

Tab 3. Tabulka směrodatných odchylek koncového bodu

2.3. Testování

Při testování modelu byl veliký důraz kladen především na správnou funkčnost výpočtu souřadnic lopaty minibagru. Ověření správnosti výsledků bylo provedeno za pomocí totální stanice *Trimble S7 robotic* s kontrolerem *TSC3* také od firmy Trimble. Přístroj byl pomocí trojnožky horizontován na nucené centraci pilíře měřické základny v laboratoři FSV ČVUT. Na lopatu modelu byl pomocí tavné pistole přilepen polystyrenový držák skrz který byl propíchnut hřebík, který sloužil k umístění držáku na požadovanou pozici. Po přilepení byl do držáku osazen mini-hranol, na který byla zacílena totální stanice. Díky tomu, že *Trimble S7* je model s poměrně vysokou výbavou, cílení neprobíhalo ručně, ale pouze za pomocí kamery integrované v přístroji ovládaným přes kontroler *TSC3*, což přineslo poměrně velikou úsporu času a ušetřilo nároky na počet lidí při testování. Pro určení stanoviska byl zvolen místní souřadný systém ve kterém byly určovány souřadnice (*X*, *Y*, *Z*) hranolu.

Obr. 10 Testování modelu ramene za pomoci Trimble S7 robotic

Testování jsem rozdělil do pěti etap, kdy v každé etapě bylo provedeno deset měření Totální stanicí a deset měření pomocí měřického systému. První dvě etapy bylo pohybováno ramenem tak, aby se měnily rozměry všech tří pístů a byl zachycen celý rozsah ramene minibagru. Souřadnicové rozdíly zobrazené v aplikaci byly zapisovány do tabulek pro následné porovnání se souřadnicemi získanými z totální stanice. Ve třetí etapě bylo pohybováno ramenem tak, aby se měnil pouze rozměr prvního pístu (d_1). Opět bylo měřeno v deseti opakováních s pokusem proměřit rovnoměrně rozsah celého pístu. Stejným způsobem byly v následujících dvou etapách otestovány i zbylé dva písty (d_2 a d_3). Tabulky s porovnáním výsledků jsou umístěny v příloze č.1 (*Testování na modelu*).

3. Tvorba aplikace pro Smartphone

V této části diplomové práce budu popisovat vývojová prostředí, která jsem si v rámci diplomové práce vyzkoušel. Také popíši aplikaci "Bagr", která je důležitou součástí tohoto projektu a slouží pro zpracování dat naměřených měřickou soustavou.

3.1. Volba cílové platformy

Před samotným vývojem je důležité rozmyslet si několik důležitých aspektů, které následně ovlivní vývoj aplikace. Nejdůležitějším aspektem je platforma, pro kterou bude aplikace vytvořena. Tu je dobré zvolit především dle cílové skupiny uživatelů. Jelikož světově nejrozšířenější platformou je *Android* od společnosti *Google*, rozhodl jsem se vytvořit aplikaci právě na tuto platformu. Aplikace jsou běžně instalovány přes obchod *Google Play*, kde je možné stáhnout jak aplikace zdarma, tak ty placené. *Android* však umožňuje instalovat aplikace také z internetu, k tomu je třeba v telefonu nastavit instalaci aplikací z cizích zdrojů. Tato volba je však riskantní a je důležité být obezřetný.

3.2. Základní dělení aplikací dle způsobu vývoje

V současnosti jsou možné tři způsoby vývoje aplikací, přičemž každý způsob slouží k jinému typu a použití mobilní aplikace.

Prvním a stále nejrozšířenějším, je nativní způsob vývoje aplikací, doporučovaný vývojáři platforem *Google (Android)*, *Apple (iOS)* a *Microsoft (Windows)*. Každá platforma má pro nativní způsob vývoje svůj programovací jazyk např. jazyk *Java*, *ObjectiveC/Swift*, *C#*.

Velmi zajímavé a důležité jsou u nativního způsobu vývoje také nástroje *Native Script* a *React Native*, které umí zkompilovat jiný zdrojový kód do nativního kódu. Nativní způsob poskytuje nejlepší výkon, avšak velkou nevýhodou je, že musí být vyvinuty pro každou mobilní platformu zvlášť, což zvyšuje náklady výsledné aplikace. S tímto problémem souvisí i opravy aplikací (každá aplikace se musí

opravovat zvlášť). Další nevýhodou jsou zvýšené nároky na programátory, kteří musí ovládat hned několik odlišných programovacích jazyků.

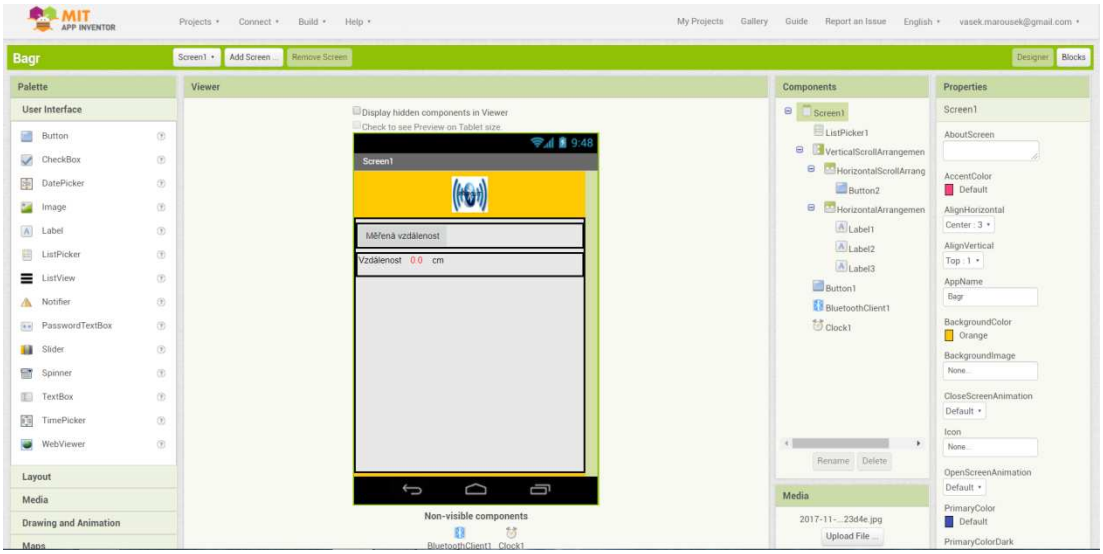
Druhým a velmi rychlým způsobem vývoje je mobilní web, využívaný tehdy, pokud by aplikace kopírovala funkcionalitu již vytvořeného a běžícího webu. Tento způsob je vhodný spíše pro nenáročné aplikace i nenáročné uživatele. Chybí zde napojení na *API* telefonu i nativní *UI* prvky, navíc uživatel musí být připojený k Internetu. Naopak velkou výhodou tohoto způsobu je univerzálnost, již zmíněná rychlost, okamžité uvedení aplikace do obchodů *Google Play* či *App Store*.

Třetím a posledním způsobem vývoje mobilních aplikací, je kombinace dvou předešlých, tzv. hybridní způsob, jehož základem je „*WebView*“ - instance prohlížeče, což znamená zobrazení webového obsahu v hybridní aplikaci za pomoci nainstalovaného prohlížeče. Velké klady vidím především v jednoduchém testování a oproti nativnímu způsobu je hybridní levnější a snazší, avšak i u této metody chybí nativní *UI* prvky, je zde možnost nedostupnosti *API* a také nároky na hardware jsou vyšší, to však v dnešní době v souvislosti se stále se zlepšujícím výkonem mobilních zařízení už zdaleka není takový problém.

[15],[16]

3.2. MIT Inventor 2

V první fázi jsem se rozhodl aplikaci vyvíjet v internetovém prostředí *MIT inventor 2* dostupném z adresy uvedené v seznamu zdrojů [8]. Toto vývojové prostředí umožňuje snadnou tvorbu aplikací pro *Android* formou interaktivního blokového schéma, které je tvořeno předpřipravenými bloky, ze kterých si pouze vybíráme. Pro práci s tímto vývojovým prostředím stačí vlastnit pouze účet *Google*, pod kterým se do webové stránky přihlásíte a můžete začít s tvorbou. Veškeré projekty jsou navíc ukládány na cloudové úložiště, a díky tomu je zajištěna vyšší bezpečnost dat. Při založení nové aplikace vložíme pouze název a můžeme začít s tvorbou.



Obr. 11 Screenshot tvorby aplikace pomocí MIT Inventor 2

V záložce *"Designer"* je tvořena grafická část aplikace. Velikou výhodou tohoto prostředí je, že prakticky ihned vidíme výslednou podobu. V *designeru* můžeme vkládat tlačítka, textová pole, ale také třeba nástroj pro připojení bluetooth a další. Aby však aplikace fungovala tak jak jsme si ji vytvořili je nutné přesunout se do záložky *"Blocks"*. Zde vytváříme zdrojový kód pomocí bloků, kde pro každou kategorii komponentů jsou připraveny nástroje a my pouze vybíráme, který z nich použijeme a graficky pomocí myši tyto bloky spojujeme. Bloky jsou spojeny až po zvukové signalizaci. Ujistit se můžete tak, že při uchopení a přesunu spojených bloků se přesouvají všechny najednou.

Při sestavení aplikace máme na výběr, zda chceme aplikaci uložit na pevný disk a nebo uložit na cloudové úložiště. Když zvolíme volbu uložení aplikace na cloudové úložiště, vygeneruje se QR kód po jehož naskenování se aplikace nainstaluje do zařízení.

Tento nástroj pro tvorbu aplikací má obrovskou výhodu v jednoduchosti a rychlosti tvorby nové aplikace. Bohužel při užití tohoto nástroje je uživatel limitován tím co již někdo vytvořil a nemá takové možnosti jako v ostatních vývojových prostředích. Právě z tohoto důvodu bylo od tohoto nástroje nakonec upuštěno a aplikace byla vytvořena ve vývojovém prostředí *Android Studio*. Při vývoji v tomto prostředí bylo využito video návodu na *Youtube*.

[9]

3.3. Android Studio

Android studio je software pro vývoj nativních aplikací od společnosti *Google*. Program je zdarma pro uživatele na platformách *Windows, Mac OS X* a *Linux*. Nabízí spoustu možností při tvorbě aplikací pro *Android*. Součástí instalačního balíčku jsou také *Android SDK tools*, kompilátor *Android* a základní emulátory s plnohodnotným systémem *Android*. Díky těmto emulátorům můžeme aplikaci otestovat i přes skutečnost, že nevlastníme žádné zařízení s tímto systémem na virtuálním zařízení přímo v počítači. Tento software navíc usnadňuje programování díky automatickému doplňování kódu jazyků *Java, Kotlin a C/C++*.

Pro ukázku práce s tímto programem si popíšeme založení nového projektu. V horní liště programu klikneme na rolovací menu a volíme následující možnosti *File >> new >> new project*, tímto vyvoláme vyskakovací okno, ve kterém zvolíme možnost *Empty Activity*. V následujícím okně je nutné vyplnit název aplikace a její umístění na disku. Dále zde zvolíme skriptovací jazyk, ve kterém chceme aplikaci vytvářet. Na výběr máme jazyk *C/ C++, Java nebo Kotlin*. Závěrem vybereme verzi systému *Android* pro kterou chceme aplikaci vytvářet. Doporučuji zvolit raději nižší verzi *Androidu* pro kompatibilitu s co možná nejvíce zařízeními. Jelikož telefon s vyšší verzí systému je schopen otevřít aplikaci vyvinutou pro nižší verze *Androidu*, zato telefon s nižší verzí *Androidu* nikoliv. Při této volbě vám navíc *Android studio* ukazuje odhad, kolik procent zařízení vámi vytvořenou aplikaci bude schopno otevřít, což je určitě pozitivní vlastnost programu.Stisknutím tlačítka *finish* se otevře vámi vytvořená aplikace a můžeme začít s tvorbou. Já jsem pro tvorbu aplikace zvolil skriptovací jazyk *Java* a verzi *Androidu* (*4.1 Jelly Bean*).

[17]

3.4. Aplikace "Bagr"

Aplikace byla vytvořena ve vývojovém prostředí *Android Studio* v jazyce *Java*. Jedná se o nativní aplikaci pro platformu *Android*. Jejími hlavními funkcemi je vyhledávání spárovaných zařízení pomocí *Bluetooth*, připojení k těmto zařízení a následný příjem dat, zpracování měřených dat měřického systému, vklad parametrů ramene a vklad směrodatných odchylek jednotlivých čidel použitých v měřickém systému. Při vývoji aplikace jsem hodně čerpal z *Youtube* od uživatele *CodingWithMitch*, který má na svém kanále obsáhlé a podrobné video návody a zdrojové soubory je možné stáhnout na portálu *Github*.

[25] [26] [27]

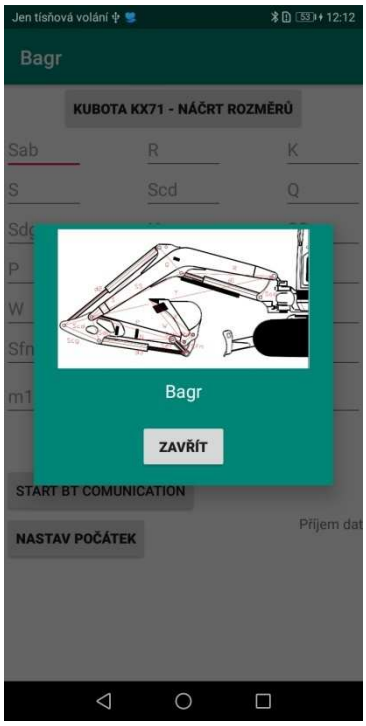
3.4.1. Úvodní obrazovka

Po spuštění aplikace se dostaneme na úvodní obrazovku, na které se nachází tlačítko *ON/OFF*. Jeho stiskem můžeme vypnout nebo naopak zapnout funkci *Bluetooth* mobilního zařízení. Pod tímto tlačítkem najdeme seznam nalezených spárovaných zařízení, kde si můžeme vybrat ke kterému zařízení se chceme připojit. Klepnutím na námi vybrané zařízení nás aplikace přesměruje na hlavní obrazovku (3.4.2.).

Obr. 12 Vedlejší obrazovka aplikace "Bagr"

3.4.2. Hlavní obrazovka

Po přesměrování na hlavní obrazovku se dostáváme k hlavním funkcím aplikace, a to k celému zpracování dat přijatých z měřického systému. V horní části obrazovky nalezneme tlačítko *"Kubota KX71 - Náčrt rozměrů"*. Po jeho stisknutí se objeví okno s obrázkem ramene minibagru, kde jsou vyznačeny jak vzdálenosti měřené měřickým systémem, tak parametry ramene, které je nutné před samotným výpočtem zadat.



Obr. 13 Hlavní obrazovka - náčrt rozměrů

Pod tímto tlačítkem se nacházejí textová pole (*EditText*), do kterých je možné zadat parametry ramene označených dle přiloženého náčrtu. Protože je aplikace vyvíjena přímo pro minibagr *Kubota KX71-3* jsou tato textová pole nastavena tak, že pokud nevložím žádná data a stisknu tlačítko *"Ulož Parametry"* uloží se do proměnných parametry ramene právě tohoto minibagru.

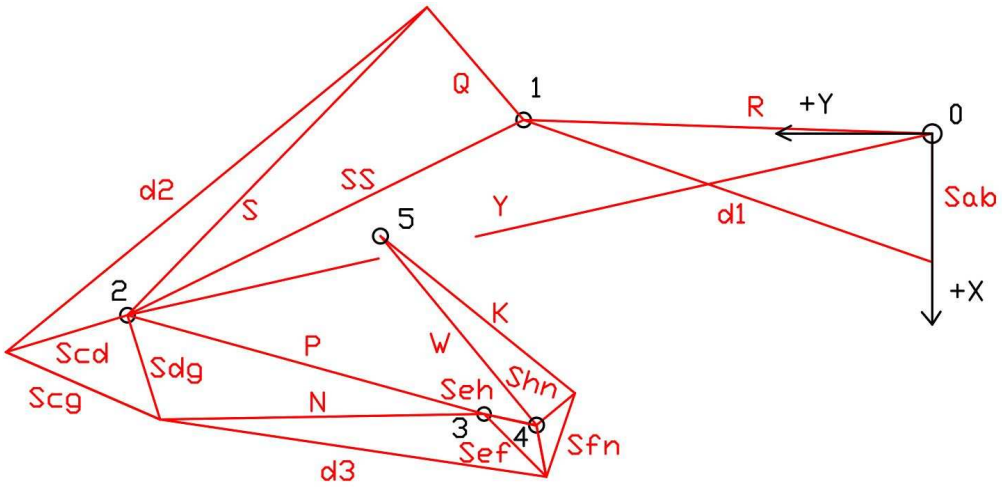
Níže se nacházejí další textová pole do kterých můžeme vložit směrodatnou odchylku vzdálenosti měřené pomocí snímače. Čísla směrodatných odchylek odpovídají číslům měřených vzdáleností (tzn. měřené délce d1 přísluší směrodatná

odchylka m1). Směrodatné odchylky uložíme stisknutím tlačítka *"Ulož parametry snímačů"*. Při každém spuštění aplikace je nutné tato dvě tlačítka stisknout, i přes to, že nevkládáme žádné parametry kvůli definici těchto parametrů.

Obr. 14 Hlavní obrazovka aplikace "Bagr"

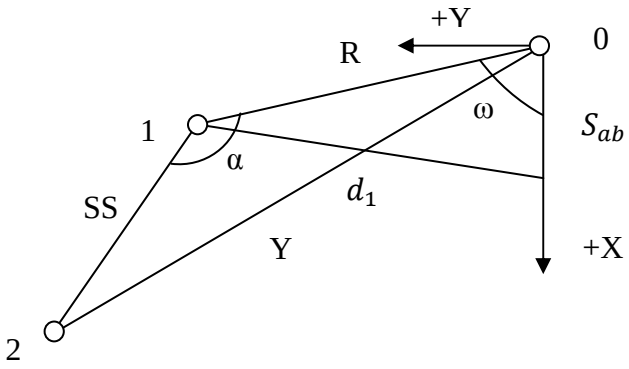
Ve spodní části hlavní obrazovky se nachází tlačítko *"Start BT comunication"*, po jeho stisknutí začne *Bluetooth* komunikace mezi *Arduinem* a mobilním zařízením což je patrné v textovém poli na pravé straně od tlačítka, kde se zobrazují měřené vzdálenosti a vypočtené souřadnicové rozdíly (postup výpočtu v podkapitole 3.5.).

3.5. Výpočet



Obr. 15 Přehled rozměrů ramene

Pro výpočet souřadnic lopaty minibagru jsem zvolil místní souřadnicový systém s počátkem v prvním kloubu ramene a kladnou větví osy X vloženou do strany S_{ab}.



Úhel ω byl vypočítán vyjádřením z kosinové věty

$$\omega = \text{acos}\left(\frac{R^2 + S_{ab}^2 - d_1^2}{2 * R * S_{ab}}\right)$$

kde R a S_{ab} jsou rozměry ramene odměřené z *Autocadu* a d_1 je velikost pístu měřená pomocí snímače.

Souřadnice Y a X bodu 1 byly vypočítány rajonem za použití počátečního směrníku ω

$$Y_1 = Y_0 + R * \sin(\omega) \quad X_1 = X_0 + R * \cos(\omega)$$

Levostranný úhel na bodě 1 byl vypočítán z pevných rozměrů vyjádřením z kosinové věty.

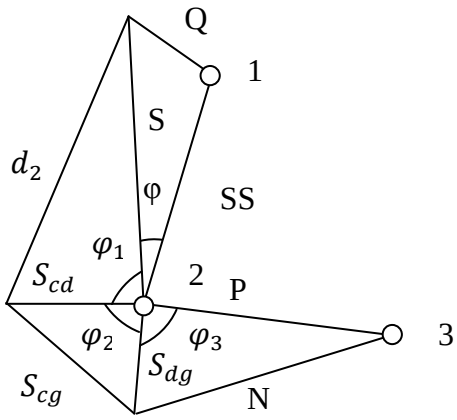
$$\alpha = \text{acos}\left(\frac{R^2 + SS^2 - Y^2}{2 * R * SS}\right)$$

kde *R*, *Y* a *SS* jsou pevné rozměry ramene.
Dále byl vypočítán směrník σ_2

$$\sigma_2 = \omega + \alpha - \pi.$$

Souřadnice bodu 2 byly vypočteny rajonem

$$Y_2 = Y_1 + SS * \sin(\sigma_2) \quad X_2 = X_1 + SS * \cos(\sigma_2)$$



Na kloubu číslo 2 bylo pro výpočet levostranného úhlu nutné nejprve vypočítat pomocné úhly φ, φ_1, φ_2 *a* φ_3. Tytp úhly byly vypočítány vyjádřením z kosinové věty.

$$\varphi = \text{acos}\left(\frac{S^2+SS^2-Q^2}{2*S*SS}\right), \quad \varphi_1 = \text{acos}\left(\frac{S^2+S_{cd}^2-d_2^2}{2*S*S_{cd}}\right), \quad \varphi_2 = \text{acos}\left(\frac{S_{cd}^2+S_{dg}^2-S_{cg}^2}{2*S_{dg}*S_{cd}}\right)$$

$$\varphi_3 = \text{acos}\left(\frac{P^2+S_{dg}^2-N^2}{2*S_{dg}*P}\right)$$

kde *S*, *SS*, *Q*, S_{cd}, S_{dg}, S_{cg}, *N* a *P* jsou pevné rozměry ramene a d_2 je velikost pístu určená měřící soustavou.

Z pomocných úhlů byl vypočítán směrník σ_3

$$\sigma_3 = \sigma_2 + (2 * \pi - \varphi - \varphi_1 - \varphi_2 - \varphi_3) - \pi$$

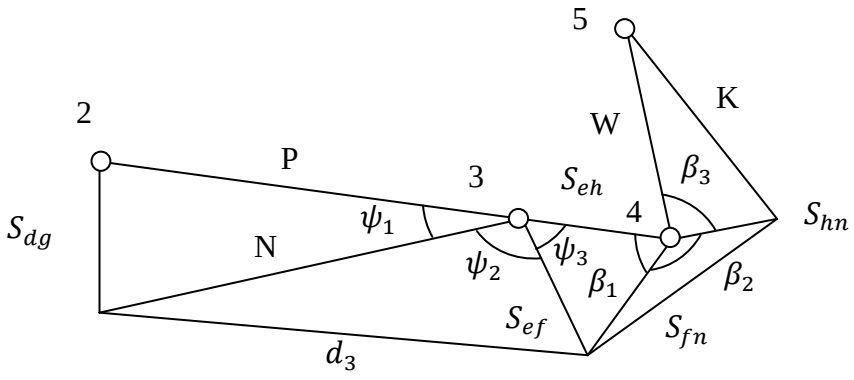
a souřadnice kloubu číslo 3 (Y_3 a X_3) které byly opět vypočítány rajonem.

$$Y_3 = Y_2 + P * \sin(\sigma_3) \quad X_3 = X_2 + P * \cos(\sigma_3)$$

Na bodě číslo 3 byly vypočítány pomocné úhly ψ_1, ψ_2 a ψ_3

$$\psi_1 = \text{acos}\left(\frac{P^2+N^2-S_{dg}^2}{2*P*N}\right), \quad \psi_2 = \text{acos}\left(\frac{N^2+S_{ef}^2-d_3^2}{2*N*S_{ef}}\right),$$

kde P, N, S_{dg} a S_{ef} jsou pevné rozměry ramene a d_3 je velikost pístu určená měřící soustavou.



Výpočet úhlu ψ_3 byl proveden pomocí podmínky *IF*, která rozlišovala dva případy které mohou nastat

$$\psi_1 + \psi_2 > \pi \quad \text{z toho} \quad \psi_3 = \psi_1 + \psi_2 - \pi$$
$$\psi_1 + \psi_2 < \pi \quad \text{z toto} \quad \psi_3 = \pi - \psi_1 - \psi_2.$$

Jelikož klouby 2, 3 a 4 leží na přímce, pro výpočet souřadnic bodu 4 bude platit, že

$$\sigma_3 = \sigma_4.$$

Souřadnice bodu 4 budou tedy vypočteny z následujících vzorců

$$Y_4 = Y_3 + S_{eh} * \sin(\sigma_4) \quad X_4 = X_3 + S_{eh} * \cos(\sigma_4).$$

Na bodě číslo 4 byly vypočítány pomocné úhly β_1, β_2 *a* β_3 a také vzdálenost S_{fh} pro výpočet levostranného úhlu polygonového pořadu a následný výpočet směrníku.

$$S_{fh} = \sqrt{S_{ef}^2 + S_{eh}^2 - 2 * S_{ef} * S_{eh} * cos(\psi_3)}$$

$$\beta_1 = \text{acos}\left(\frac{W^2+S_{hn}^2-K^2}{2*W*S_{hn}}\right), \quad \beta_2 = \text{acos}\left(\frac{S_{fh}^2+S_{hn}^2-S_{fn}^2}{2*S_{fh}*S_{hn}}\right), \quad \beta_3 = \text{acos}\left(\frac{S_{cd}^2+S_{dg}^2-S_{cg}^2}{2*S_{dg}*S_{cd}}\right)$$

kde W, K, S_{cd}, S_{dg}, S_{cg}, S_{fh}, S_{hn}, S_{fn} jsou pevné rozměry ramene.

Směrník

$$\sigma_5 = \sigma_4 + (2 * \pi - \beta_1 - \beta_2 - \beta_3) - \pi$$

Nakonec byly vypočítány souřadnice zubu lopaty

$$Y_5 = Y_4 + W * \sin(\sigma_5) \quad X_4 = X_4 + W * \cos(\sigma_5).$$

V aplikaci jsou zobrazovány souřadnicové rozdíly *diferenceY* a *diferenceX*.

$$diferenceY = Y0 - Y5 \quad diferenceX = X0 - X5$$

kde Y0 a X0 jsou souřadnice zubu lopaty Y5 a X5 vypočítané při prvním přijatém měření přes *Bluetooth*, nebo první vypočtené souřadnice po stisku tlačítka "Nastav počátek".

4. Osazení systému na minibagru

V této části diplomové práce uvedu základní informace o minibagru, na kterém byla měřická soustava testována**.** Také zde bude popsáno testování.

4.1. Kubota KX 71-3

Kubota KX 71-3 je mini rýpadlo s vysokým výkonem, výbornou zvedací silou a palivovým motorem *V1505-EBH*. S ramenem dlouhým *1350 mm* a kvalitní lopatou, dokáže kopat až *2680 mm* hluboko, což je skoro o *20 cm* více než hloubka výkopu příbuzného modelu *KX 61-3*.

Pro efektivní provoz mini rýpadla se průtok a tlak oleje nastavují podle pracovní zátěže, k tomu se používá variabilní čerpadlo. Motor pracuje ekonomičtěji a díky tomu vydrží s plnou nádrží cca *10* hodin nepřetržité práce. *Kubota KX 71-3* má také vlastní inteligentní řídicí systém vybavený kontrolkami pro motor, palivo, olej, teplotu- pro přesné určení problému, dále také standardní ukazatele zobrazující aktuální pracovní podmínky: otáčkoměr, měřič hodin. Maximální rychlost pojezdu je 2,7 km/h.

4.2. Testování

Nejprve byl minibagr osazen snímači a to opět pomocí úchytů vyříznutých z polystyrenu. Před lepením byly plochy očištěny odmašťovacím prostředkem aby mohli být úchyty přilepeny pomocí oboustranné lepící pásky. Kvůli ochraně před deštěm a zpevněním celého osazení byly snímače ještě přelepeny lepící páskou tak aby nebránila měření. Na lopatu byl umístěn odrazný hranol pro ověření polohy. Funkčnost celého systému byla ověřena totální stanicí *Leica TC1700*, kterou byly měřeny vodorovné úhly, zenitové úhly a šikmé délky, které byly ukládány do paměti přístroje. Zároveň byl prováděn zápis souřadnicových rozdílů vypočtených aplikací. Výpočet souřadnic byl proveden v *Gromě* polární metodou včetně souřadnice Z, ze které byly vypočtena převýšení.

Obr. 16 Testování měřícího systému na minibagru

Výsledné souřadnice byly zapsány do tabulky, kde je ověřena přesnost měřického systému. Tato tabulka je součástí přílohy č. 2 (*Testování na minibagru*)

5. Závěr

Při tvorbě této práce jsem otestoval několik různých typů snímačů pro měření vzdálenosti. Ověřil jsem zda přesnost udávaná výrobcem odpovídá skutečnosti a také zda je použití těchto snímačů možné i v praxi. Testováním bylo prokázáno, že po úpravě došlo ke snížení rozptylu měřených hodnot, zvýšení přesnosti a také především zvýšení dosahu. Díky této skutečnosti byly pro celý systém nakonec použity pouze tyto snímače a to především díky nízké pořizovací ceně.

V rámci této diplomové práce jsem vytvořil aplikaci pro chytrý telefon s platformou *Android*. Aplikace byla vytvořena pomocí vývojového prostředí Android Studio v jazyce Java. Právě pomocí této aplikace je měřící systém ovládán a naměřená data jsou zde přijata pomocí *Bluetooth* komunikace mezi *Arduinem* a chytrým telefonem. Výpočet výškových rozdílů je prováděn aplikací z důvodu vyššího výkonu telefonu oproti *Arduinu*. Aplikace je navržena tak, aby do ni bylo možné vložit parametry jiných bagrů a směrodatné odchylky snímačů. Výsledky jsou zobrazovány v dolní části hlavní obrazovky. Kvůli kontrole správného pořadí snímačů jsou zobrazovány i měřené délky.

Dalším krokem byl výpočet modelované přesnosti koncového bodu polygonového pořadu kvůli výpočtu mezního rozdílu před testováním.

Po vývoji aplikace a sestavení měřícího systému byl vytvořen model, na kterém bylo provedeno testování. Nejprve byly takto odstraněny chyby v aplikaci a ve výpočtu a poté byla testována samotná přesnost měřícího systému. Při testování nebyl překročen mezní rozdíl a tím byla prokázána přesnost vypočtená modelací.

Kvůli ověření reálného využití tohoto systému bylo provedeno testování na minibagru *Kubota KX71-3.* Při testování byla také prokázána modelovaná přesnost a ověřena správná funkčnost celého systému v reálných podmínkách.

5. Závěr

Při použití ultrazvukových dálkoměrů byla dosažena modelovaná směrodatná odchylka polohy lopaty minibagru *75.7 mm.* Tato hodnota by se dala zlepšit například použitím snímače HI50.

Tabulky s přehledem výsledků testování jsou součástí přílohy. Zdrojové kódy pro *Arduino* a výpočetní skripty z programu *Matlab* jsou součástí elektronických příloh. Zdrojový projekt aplikace *"Bagr"* je také součástí elektronických příloh.

V dalším vývoji celého systému bych se zaměřil především na řešení problému s velkým množstvím kabelů, které by při reálném použití překáželi. Z toho důvodu by bylo lepší osadit bagr třemi *Arduiny*, přičemž na každé by byl napojen jeden snímač vzdálenosti a jeden *Bluetooth* modul, a zařízení by komunikovala mezi sebou. Jedno z těchto zařízení by posílalo data do chytrého telefonu. Tento krok by však zvedl náklady na celý systém. Dalším krokem kupředu by bylo přidání tlačítek pomocí kterých by bylo možné volit mezi lopatami dle jejich velikosti. snížil by se počet vkládaných rozměrů a aplikace by se stala uživatelsky příjemnější. Přidáním snímače pro detekci úhlu otočení by se navíc dala do výpočtu přidat k výškové složce ta polohová. Zde by však stálo za zvážení zda by nebylo výhodnější osadit na rameno modul pro příjem GNSS dat.

6. Seznam zdrojů

[1] *GY-VL53L0X* [online]. www.st.com, 2016 [cit. 2019-04-29]. Dostupné z: https://drive.google.com/file/d/0B4B30jzMyzG8SnZtZHNGSlAtaVBmbjJXV3J3OFhlSzlITEkw/view

[2] GY-VL53L0X - zdroj kódu [online]. Oleglyan, 2016 [cit. 2019-04-29]. Dostupné z: https://forum.pololu.com/t/vl53l0x-maximum-sensors-on-i2c-arduino-bus/10845/7

[3] *HC-SR04* [online]. Luboš M., 2016 [cit. 2019-04-29]. Dostupné z: https://navody.arduino-shop.cz/navody-k-produktum/meric-vzdalenosti-ultrazvukovy.html

[4] *HC-SR04_datasheet* [online]. iteadstudio.com, 2010 [cit. 2019-04-29]. Dostupné z: https://www.electroschematics.com/wp-content/uploads/2013/07/HC-SR04-datasheet-version-2.pdf

[5] *HC-SR04 - knihovna NewPing* [online]. 2016 [cit. 2019-04-29]. Dostupné z: https://playground.arduino.cc/Code/NewPing/

[6] *DHT11 - ovládání senzoru* [online]. Luboš M., 2016 [cit. 2019-04-29]. Dostupné z: https://navody.arduino-shop.cz/navody-k-produktum/teplotni-senzor-dht11.html

[7] *DHT11 - datasheet* [online]. aosong, [cit. 2019-04-29]. Dostupné z: https://akizukidenshi.com/download/ds/aosong/DHT11.pdf

[8] http://ai2.appinventor.mit.edu/?locale=en#6292563355631616

[9] How to MIT APP INVENTOR 2 Make a Bluetooth Connection Screen [online]. Z-HUT, 2017 [cit. 2019-05-10]. Dostupné z: https://www.youtube.com/watch?v=F9SzF3ktkIo

[10] Návrh a konstrukce vzdáleného experimentu [online]. Olomouc, 2015 [cit. 2019-05-10]. Dostupné z: https://theses.cz/id/wajj2j/Jakub_Doskoil_Nvrh_a_konstrukce_vzdlenho_experimentu_-_me.pdf

[11] DHT11 - picture [online]. amazon [cit. 2019-05-10]. Dostupné z: https://www.amazon.com/DHT11-Digital-Temperature-Humidity-Sensor/dp/B00V2DWL2E

[12] Arduino Mega 2560 [online]. amazon [cit. 2019-05-10]. Dostupné z: https://www.amazon.com/SmartProjects-Mega-Arduino-MEGA-Board/dp/B004A7H3DG

[13] Trimble S7 robotic datasheet [online]. Trimble, 2015 [cit. 2019-05-10]. Dostupné z: https://geospatial.trimble.com/sites/default/files/2019-03/Datasheet%20-%20Trimble%20S7%20Total%20Station%20-%20Czech%20-%20Screen.pdf

[14] *Leica TC1700 datasheet* [online]. Leica, 2016 [cit. 2019-05-10]. Dostupné z: https://www.propertyandlandtitles.vic.gov.au/__data/assets/pdf_file/0032/54968/Leica-EDM-Instrument-Parameters.pdf

[15] *Vývoj multiplatformních mobilních aplikací* [online]. Pavel Mašek, 2017 [cit. 2019-05-10]. Dostupné z: https://theses.cz/id/20k2p8/STAG89569.pdf

[16] *Jak vyvíjet mobilní aplikace* [online]. Jan Václavík [cit. 2019-05-10]. Dostupné z: http://janvaclavik.cz/jak-vyvijet-mobilni-aplikace/

[17] Android studio [online]. [cit. 2019-05-10]. Dostupné z: https://developer.android.com/studio/?gclid=Cj0KCQjwt_nmBRD0ARIsAJYs6o0pClZXXTqSR93LQ3Uhg9ddYX-v2Id7FpGt6MAfhgXht0t3jy7HPJcaAgPFEALw_wcB

[18] Třídy laserů [online]. Leonardo Technology [cit. 2019-05-10]. Dostupné z: http://www.lt.cz/e-learning/laser/bezpecnost-laseru-tridy-1-az-4

[19] GY-VL53L0X [online]. Smart Prototyping [cit. 2019-05-10]. Dostupné z: https://www.smart-prototyping.com/VL53L0X-Time-of-Flight-Distance-Sensor-GY-530

[20] *HC-SR04* [online]. Andrew Fisher, 2015 [cit. 2019-05-18]. Dostupné z: http://johnny-five.io/news/hcsr04-i2c-backpack-support-lands/

[21] HI50 [online]. Aliexpress [cit. 2019-05-18]. Dostupné z: https://www.aliexpress.com/item/50m-164ft-Laser-Distance-Measuring-Sensor-Range-Finder-Module-Low-cost-Diastimeter-Single-Continuous-Measurement/32824312275.html?spm=2114.search0604.3.8.6ca74de86FEQbd&ws_ab_test=searchweb0_0,searchweb201602_8_10065_10068_319_10059_10884_317_10887_10696_321_322_10084_453_10083_454_10103_10618_10304_10307_10820_10821_537_10302_536,searchweb201603_52,ppcSwitch_0&algo_expid=c24985e8-01c6-4cdd-aa76-b4e1004c07b5-1&algo_pvid=c24985e8-01c6-4cdd-aa76-b4e1004c07b5

[22] Vl53L0X.h [online]. Pololu, 2017 [cit. 2019-05-18]. Dostupné z: https://github.com/pololu/vl53l0x-arduino/blob/master/README.md

[23] *Bluetooth modul HC-05* [online]. Luboš M., 2017 [cit. 2019-05-18]. Dostupné z: https://navody.arduino-shop.cz/navody-k-produktum/arduino-bluetooth-modul-hc-05.html

[24] Tištěný datasheet dodaný výrobcem

[25] Android studio - Bluetooth návod [online]. CodingWithMitch, 2016 [cit. 2019-05-18]. Dostupné z:https://www.youtube.com/watch?v=y8R2C86 BIUc&list=PLgCYzUzKIBE8K HMzpp6JITZ2JxTgWqDH2

[26] Android studio - Github zdrojové kódy [online]. Mitch Tabian, 2016 [cit. 2019-05-18]. Dostupné z: https://github.com/mitchtabian

[27] Android studio - popis a vyhledávání funkcí vývojového prostředí AS [online]. Developers Android [cit. 2019-05-18]. Dostupné z: https://developer.android.com/

[28] *Ikona bluetooth v aplikaci* [online]. Kungfuoctopus's blog, 2009 [cit. 2019-05-18]. Dostupné z:https://kungfuoctopus.files.wordpress.com/2010/ 01/bluetooth-icon-custom.png

7. Seznam obrázků

Obr. 1: Laserový snímač vzdálenosti "GY-VL53L0X" [19] 9
Obr. 2: Graf měřených hodnot "GY-VL53L0X" 11
Obr. 3: Ultrazvukový snímač vzdálenosti "HC-SR04" [20] 13
Obr. 4: Testování snímačů - délková základna FSV ČVUT 16
Obr. 5: Snímač DHT11 [11] 17
Obr. 6: Arduino Mega 2560 [12] 18
Obr. 7: Snímač vzdálenosti HI50 [21] 21
Obr. 8: Bluetooth modul HC-05 [23] 22
Obr. 9: Model ramene minibagru 24
Obr. 10:Testování modelu ramene za pomoci Trimble S7 robotic 26
Obr. 11:Screenshot tvorby aplikace pomocí MIT Inventor 2 29
Obr. 12:Vedlejší obrazovka aplikace "Bagr" 32
Obr. 13:Hlavní obrazovka - náčrt rozměrů 33
Obr. 14:Hlavní obrazovka aplikace "Bagr" 34
Obr. 15:Přehled rozměrů ramene 35
Obr. 16:Testování měřícího systému na minibagru 40

8. Seznam tabulek

Tab. 1: Test tří snímačů GY-VL53L0X současně 12

Tab. 2: Test úpravy ultrazvukového dálkoměru 16

Tab. 3: Tabulka směrodatných odchylek koncového bodu 25

9. Seznam schémat zapojení

Schéma 1: GY-VL53L0X 10
Schéma 2: Test tří snímačů GY-VL53L0X současně 11
Schéma 3: HC-SR04 + DHT11 14
Schéma 4: 2 x HC-SR04 + DHT11 15
Schéma5: Kompletní zapojení měřického systému 23

10. Seznam zkratek

AS	Android Studio
MIT inventor	Massachusetts Institute of Technology
API	Aplication programming interface
UI	User interface
FSV	Fakulta stavební
ČVUT	České vysoké učení technické

11. Seznam příloh

Příloha 1: Testování na modelu
Příloha 2: Testování na minibagru

Elektronicky:
Příloha 3: Zdrojový kód Arduino testování VL53L0X
Příloha 4: Výpočet přesnosti VL53L0X
Příloha 5: Zdrojový kód Arduino testování 3xVL53L0X
Příloha 6: Výpočet přesnosti 3x VL53L0X
Příloha 7: Složka s grafy měřených veličin pomocí tří laserových dálkoměrů
Příloha 8: Ovládání TS a snímačů
Příloha 9: Zdrojový kód Arduino testování HC-SR04 + DHT11
Příloha 10: Zdrojový kód Arduino Testování 2xHC-SR04 + DHT11
Příloha 11: Výpočet směrodatných odchylek snímačů na základě testování fáze 2
Příloha 12: Výpočet přesnosti koncového bodu
Příloha 13: Zdrojový kód Arduino pro ovládání měřického systému
Příloha 14: Složka souborů aplikace "Bagr"

Příloha č.1

Testování na modelu

Etapa č. 1								
Číslo bodu	Souřadnicové rozdíly určené pomocí měřické soustavy		Souřadnice určené totální stanicí			Převýšení TS	Rozdíl převýšení	Mezní rozdíl
	ΔY'	ΔX' = ΔZ'	Y	X	Z	ΔZ	ΔZ-ΔZ'	
	[m]	*[m]*	*[m]*	*[m]*	*[m]*	*[m]*	*[m]*	*[m]*
1	0,000	0,000	8,356	50,441	1,565	0,000	0,000	0,107
2	-0,217	0,026	8,323	50,352	1,545	-0,020	0,006	
3	-0,271	0,199	8,299	50,296	1,313	-0,252	-0,053	
4	-0,285	-0,365	8,301	50,240	1,890	0,325	-0,040	
5	-0,049	-0,219	8,340	50,457	1,837	0,272	0,053	
6	0,163	-0,176	8,359	50,583	1,780	0,215	0,039	
7	0,160	0,012	8,316	50,411	1,543	-0,022	-0,010	
8	-0,198	-0,206	8,317	50,370	1,801	0,236	0,030	
9	-0,133	-0,228	8,302	50,286	1,809	0,244	0,016	
10	-0,340	0,101	8,306	50,258	1,446	-0,119	-0,018	

Etapa č. 2								
Číslo bodu	Souřadnicové rozdíly určené pomocí		Souřadnice určené totální stanicí			Převýšení TS	Rozdíl převýšení	Mezní rozdíl
	ΔY'	ΔX' = ΔZ'	Y	X	Z	ΔZ	ΔZ-ΔZ'	
	[m]	*[m]*	*[m]*	*[m]*	*[m]*	*[m]*	*[m]*	*[m]*
1	0,000	0,000	8,318	50,479	1,546	0,000	0,000	0,107
2	-0,070	0,151	8,301	50,398	1,343	-0,203	-0,052	
3	0,061	-0,107	8,317	50,491	1,702	0,156	0,049	
4	-0,058	-0,402	8,309	50,386	1,919	0,373	-0,029	
5	0,091	-0,282	8,302	50,436	1,881	0,335	0,053	
6	0,284	-0,082	8,310	50,495	1,641	0,095	0,013	
7	-0,400	0,149	8,285	50,408	1,393	-0,153	-0,004	
8	-0,282	-0,112	8,278	50,254	1,618	0,072	-0,040	
9	0,111	-0,177	8,310	50,469	1,775	0,229	0,052	
10	-0,081	-0,318	8,280	50,260	1,915	0,369	0,051	

Etapa č. 3								
Číslo bodu	Souřadnicové rozdíly určené pomocí		Souřadnice určené totální stanicí			Převýšení TS	Rozdíl převýšení	Mezní rozdíl
	ΔY'	ΔX' = ΔZ'	Y	X	Z	ΔZ	ΔZ-ΔZ'	
	[m]	*[m]*	*[m]*	*[m]*	*[m]*	*[m]*	*[m]*	*[m]*
1	0,000	0,000	8,334	50,513	1,515	0,000	0,000	0,107
2	-0,098	0,096	8,322	50,455	1,418	-0,097	-0,001	
3	-0,233	0,260	8,289	50,342	1,246	-0,269	-0,009	
4	-0,117	0,133	8,311	50,430	1,395	-0,120	0,013	
5	-0,011	-0,131	8,328	50,514	1,632	0,117	-0,014	
6	0,128	-0,177	8,367	50,617	1,677	0,162	-0,015	
7	0,203	-0,284	8,367	50,643	1,797	0,282	-0,002	
8	0,128	-0,142	8,361	50,603	1,637	0,122	-0,020	
9	0,045	-0,041	8,345	50,548	1,545	0,030	-0,011	
10	-0,117	0,096	8,302	50,451	1,417	-0,098	-0,002	

Etapa č. 4								
Číslo bodu	Souřadnicové rozdíly určené pomocí		Souřadnice určené totální stanicí			Převýšení TS	Rozdíl převýšení	Mezní rozdíl
	ΔY'	ΔX' = ΔZ'	Y	X	Z	ΔZ	ΔZ-ΔZ'	
	[m]	*[m]*	*[m]*	*[m]*	*[m]*	*[m]*	*[m]*	*[m]*
1	0,000	0,000	8,373	50,509	1,840	0,000	0,000	0,107
2	-0,146	-0,181	8,339	50,331	2,028	0,188	0,007	
3	-0,149	-0,286	8,333	50,294	2,126	0,286	0,000	
4	0,026	0,016	8,389	50,603	1,803	-0,037	-0,021	
5	0,143	0,093	8,413	50,706	1,788	-0,052	0,041	
6	-0,009	0,059	8,383	50,576	1,812	-0,028	0,031	
7	-0,160	-0,118	8,347	50,382	1,950	0,110	-0,008	
8	-0,149	-0,272	8,332	50,288	2,155	0,315	0,043	
9	-0,084	-0,131	8,348	50,390	1,940	0,100	-0,031	
10	-0,114	-0,101	8,359	50,441	1,889	0,049	-0,052	

Etapa č. 5								
Číslo bodu	Souřadnicové rozdíly určené pomocí		Souřadnice určené totální stanicí			Převýšení TS	Rozdíl převýšení	Mezní rozdíl
	ΔY'	ΔX' = ΔZ'	Y	X	Z	ΔZ	ΔZ-ΔZ'	
	[m]	*[m]*	*[m]*	*[m]*	*[m]*	*[m]*	*[m]*	*[m]*
1	0,000	0,000	8,376	50,517	1,874	0,000	0,000	0,107
2	-0,018	-0,075	8,378	50,546	1,961	0,087	0,012	
3	-0,018	-0,141	8,386	50,579	2,000	0,126	-0,015	
4	0,013	-0,177	8,388	50,600	2,013	0,139	-0,038	
5	0,079	0,069	8,358	50,554	1,778	-0,096	-0,027	
6	0,043	0,049	8,352	50,537	1,812	-0,062	-0,013	
7	0,000	0,000	8,354	50,527	1,875	0,001	0,001	
8	-0,054	-0,064	8,362	50,529	1,950	0,076	0,012	
9	-0,021	-0,093	8,369	50,554	1,982	0,108	0,015	
10	-0,008	-0,141	8,389	50,618	2,023	0,149	0,008	

Příloha č.2

Testování na minibagru

Testování měřícího systému na minibagru Kubota KX-71								
Číslo bodu	Souřadnicové rozdíly určené pomocí měřické soustavy		Souřadnice určené totální stanicí			Převýšení TS	Rozdíl převýšení	Mezní rozdíl
	ΔY'	ΔX' = ΔZ'	Y	X	Z	ΔZ	ΔZ-ΔZ'	
	[m]	*[m]*	*[m]*	*[m]*	*[m]*	*[m]*	*[m]*	*[m]*
1	0,000	0,000	95,572	499,113	-0,829	0,000	0,000	0,107
2	-0,327	0,074	95,850	499,082	-0,859	-0,030	0,044	
3	-0,632	-0,071	96,235	499,030	-0,779	0,050	-0,021	
4	-0,928	-0,163	96,657	498,972	-0,642	0,187	0,024	
5	-1,334	-0,119	97,054	498,914	-0,684	0,145	0,026	
6	-1,424	-0,169	97,392	498,867	-0,630	0,199	0,030	
7	-1,521	-0,258	97,684	498,834	-0,606	0,223	-0,035	
8	-1,183	-0,276	97,370	498,875	-0,570	0,259	-0,017	
9	-1,900	0,223	97,465	498,863	-1,095	-0,266	-0,043	
10	-1,311	0,380	96,945	498,932	-1,249	-0,420	-0,040	
11	-1,153	0,430	96,422	499,004	-1,314	-0,485	-0,055	
12	-0,678	0,490	95,720	499,102	-1,343	-0,514	-0,024	
13	-0,139	0,486	95,285	499,164	-1,379	-0,550	-0,064	
14	-0,026	0,743	95,204	499,171	-1,615	-0,786	-0,043	
15	0,546	-0,595	95,274	499,159	-0,195	0,634	0,039	
16	0,071	-0,259	95,489	499,127	-0,523	0,306	0,047	
17	-0,009	-0,259	95,499	499,125	-0,542	0,287	0,028	
18	0,133	-0,240	95,511	499,124	-0,562	0,267	0,027	
19	0,071	-0,191	95,522	499,122	-0,585	0,244	0,053	
19a	0,071	-0,191	95,526	499,121	-0,590	0,212	0,021	
20	-0,019	-0,793	95,543	499,117	-0,617	0,829	0,036	