Podatkovna rukavica za snimanje i analizu pokreta

Štivić, Ivan

Master's thesis / Diplomski rad

2019

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Mechanical Engineering and Naval Architecture / Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:235:984807

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-03-29**

Repository / Repozitorij:

Repository of Faculty of Mechanical Engineering and Naval Architecture University of Zagreb







SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

DIPLOMSKI RAD

Ivan Štivić

Zagreb, 2019.

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

DIPLOMSKI RAD

Mentor:
Prof. dr.sc. Zoran Kunica, dipl.ing.

Student:
Ivan Štivić

Zagreb, 2019.

IZJAVA

Izjavljujem da sam ovaj rad izradio samostalno koristeći znanja stečena tijekom studija i navedenu literaturu.

Zahvaljujem mentoru prof. dr.sc. Zoranu Kunici na pruženoj motivaciji, korisnim savjetima, uputama, kritikama te nabavljenoj opremi u svrhu realizacije ovog rada. Ništa manje zahvaljujem i asistentu Damiru Beliću, mag.ing. mech. na trudu, ukazanoj pomoći, pruženim savjetima i idejama.

Zahvaljujem svojoj djevojci Luciji, na pružanoj ljubavi, nesebičnoj potpori, konstantnom razumijevanju i toleranciji unatoč mojoj okupiranosti oko izrade ovog rada.

Zahvaljujem svojim roditeljima, majci Božici i ocu Željku te sestrama Nikolini i Mariji za sve što su učinili kako bi mi olakšali studij.

Zahvaljujem i svima ostalima koji su na bilo koji način pomogli i pridonijeli u izradi ovog rada.

U Tuheljskim Toplicama, 18. ožujka 2019.

_____

Ivan Štivić

ZADATAK

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE
Središnje povjerenstvo za završne i diplomske ispite
Povjerenstvo za diplomske radove studija strojarstva za smjerove:
proizvodno inženjerstvo, računalno inženjerstvo, industrijsko inženjerstvo i menadžment,
inženjerstvo materijala te mehatronika i robotika

Sveučilište u Zagrebu Fakultet strojarstva i brodogradnje	
Datum:	Prilog:
Klasa:	
Ur. broj:	

DIPLOMSKI ZADATAK

Student: **IVAN ŠTIVIĆ** Mat. br.: 0016082543

Naslov rada na hrvatskom jeziku: **Podatkovna rukavica za snimanje i analizu pokreta**

Naslov rada na engleskom jeziku: **Data glove for motion capture and analysis**

Opis zadatka:

Primjena napredne senzorike omogućava sve kvalitetnije poimanje gibanja i pokreta ljudskog tijela, kao još uvijek najrasprostranjenijih i neizostavnih u množini radnih procesa, a kao osnovu za planiranje i optimiranje izvedbe ručnih procesa (stabilizacija i normiranje radnih mjesta), ali i razvoj mehatroničkih uređaja s ciljem automatske izvedbe procesa.

U radu je potrebno:
1. navesti značaj snimanja pokreta za projektiranje radnih procesa i opisati trenutačni stupanj razvijenosti tehnologije
2. razviti podatkovnu rukavicu za snimanje pokreta
3. provesti eksperimente snimanja pokreta i analizirati rezultate
4. naznačiti mogućnosti povezivanja rezultata eksperimenata s raznolikim radnim procesima i njihovim planiranjem (naprimjer: radnje montaže, demontaže i ostale u vezi industrijskih proizvoda, ali i u vezi kirurški minimalno invazivnih i ORL zahvata).

Redovito se konzultirati s mentorom i asistentom D. Belićem, mag.ing. mech.

Zadatak zadan:	Rok predaje rada:	Predviđeni datum obrane:
17. siječnja 2019.	21. ožujka 2019.	27. ožujka 2019. 28. ožujka 2019. 29. ožujka 2019.

Zadatak zadao: prof. dr. sc. Zoran Kunica

Predsjednica Povjerenstva: prof. dr. sc. Biserka Runje

SADRŽAJ

IZJAVA I
ZADATAK II
POPIS OZNAKA I MJERNIH JEDINICA FIZIKALNIH VELIČINA VI
POPIS SLIKA VII
POPIS TABLICA X
SAŽETAK XI
SUMMARY XII

1. UVOD 1

2. STUDIJ POKRETA 3

3. SUSTAVI UNAPRIJED ODREĐENIH VREMENA 6

3.1. Sustav MTM i njegov razvoj 7

3.2. Sustav MTM-1 9

3.3. Sustav MTM-2 11

3.4. Označavanje i način zapisivanja pokreta u sustavu MTM-2 18

3.5. Sile pri pokretima 19

4. PODATKOVNE RUKAVICE 21

4.1. Podatkovna rukavica Sandinea i Defantija – *The Sayre glove* 21

4.2. Zimmermanova podatkovna rukavica – *The data glove* 22

4.3. Rukavica *Nintendo power glove* 23

4.4. Rukavica *Cyberglove* 25

4.5. Rukavica *P5 dataglove* 28

4.6. Rukavica *WearHap* 29

4.7. Primjena podatkovnih rukavica 30

5. OSJETILA PODATKOVNE RUKAVICE 32

5.1. Senzori pozicije šake 32

5.1.1. Senzori na mehaničkoj konstrukciji 32

5.1.2. Optički senzori 33

5.1.3. Akustički senzori 34

5.1.4. Elektromagnetski senzori 34
5.1.5. Kapacitivni senzori....35

5.2. Senzori orijentacije....35
5.3. Senzori savijanja prstiju 35
5.4. Senzori sile 36
5.5. Taktilni senzori....36
5.5.1. Otpornički taktilni senzori 38
5.5.2. Kapacitivni taktilni senzori....38
5.5.3. Piezoelektrični taktilni senzori....38
5.5.4. Ultrazvučni taktilni senzori....39
5.5.5. Optički taktilni senzori 39

6. RAZVOJ VLASTITE PODATKOVNE RUKAVICE....40
6.1. Taktilni senzori vlastite podatkovne rukavice....41
6.1.1. Električna svojstva pojedinih slojeva senzora 47
6.1.2. Električna svojstva senzora....47
6.1.3. Slika odziva senzora na računalu 48

6.2. Sastavljanje podatkovne rukavice 56
6.3. Interpretacija rada rukavicom na računalu....61
6.3.1. Uspostavljanje komunikacije....61
6.3.2. Čitanje i ispisivanje vrijednosti senzora 64
6.3.3. Kalibracija senzora i računanje vrijednosti opterećenja....67

7. EKSPERIMENTI PODATKOVNOM RUKAVICOM....72
7.1. Snimanje pokreta opisanih MTM sustavom 72
7.1.1. Pokret uzimanja, slučaj GA 72
7.1.2. Pokret uzimanja, slučaj GB 76
7.1.3. Pokret uzimanja, slučaj GC 78

7.2. Snimanje nekih učestalih pokreta....83
7.2.1. Klikanje na mišu 84
7.2.2. Uvrtanje i odvrtanje čepa boce 87
7.2.3. Umetanje i vađenje USB *sticka* 90

7.3. Zaključivanje o rukovanom objektu 93

7.3.1. Držanje prazne čaše vode 93
7.3.2. Držanje čaše ispunjene vodom....96
7.3.3. Držanje valjkastih predmeta 98

8. ZAKLJUČAK 106

9. LITERATURA....109

10. PRILOG 111

10.1. Programski kôd u programu IDE 111

10.1.1. Očitavanje i ispis vrijednosti senzora....111
10.1.2. Očitavanje i ispis vrijednosti u suradnji s MATLAB-om 112

10.2. Programski kôd u MATLAB-u 113

10.2.1. Program za animacijski prikaz vrijednosti senzora....113
10.2.2. Program za ispis iscrtavanje vrijednosti na grafu 118
10.2.3. Program za kalibriranje i aproksimaciju na izmjerenim vrijednostima 121
10.2.4. Programski kôd funkcije: Uspostavljanje_veze....123
10.2.5. Programski kôd funkcije: zatvoriPort 123
10.2.6. Programski kôd funkcije: Ocitaj_vrijednost_palac....123
10.2.1. Programski kôd funkcije: Ocitaj_vrijednost_kaziprst....124
10.2.2. Programski kôd funkcije: Ocitaj_vrijednost_rednji 124
10.2.3. Programski kôd funkcije: Ocitaj_vrijednost_prstenjak....124
10.2.4. Programski kôd funkcije: Ocitaj_vrijednost_mali 124

POPIS OZNAKA I MJERNIH JEDINICA FIZIKALNIH VELIČINA

Oznaka	Naziv	Mjerna jedinica
F	Sila	N
I	Električna struja	A
M	Moment	Nm
m	Masa	kg
p	Tlak	Pa
R	Električni otpor	Ω
t	Vrijeme	s
U	Električni napon	V

POPIS SLIKA

Slika 1. Pravilno bacanje loptice u *baseballu* [1] 2
Slika 2. Pokret uzimanje G [4] 12
Slika 3. Slučaj uzimanja GA [4] 13
Slika 4. Slučaj uzimanja GB [4] 13
Slika 5. Slučaj uzimanja GC [4] 14
Slika 6. Pokret postavljanje P [4] 14
Slika 7. Slučaj postavljanja PA [4] 15
Slika 8. Slučaj postavljana PB [4] 15
Slika 9. Slučaj postavljanja PC [4] 16
Slika 10. Pokret ponovno hvatanje R [4] 17
Slika 11. Pokret pritiskivanje A [4] 17
Slika 12. Pokret okretanje ručice C [4] 18
Slika 13. Primjer zapisivanja pokreta u sustavu MTM-2 18
Slika 14. Podatkovna rukavica Sandinea i Defantija – *The Sayre glove* [19] 22
Slika 15. Zimmermanova podatkovna rukavica – *The data glove* [20] 23
Slika 16. Rukavica *Nintendo power glove* [8] 24
Slika 17. Rukavica *CyberGlove II* [9] 26
Slika 18. Rukavica *CyberGlove III* [9] 26
Slika 19. Rukavica *CyberTouch* [9] 27
Slika 20. Rukavica *CyberGrasp* [9] 27
Slika 21. Rukavica *CyberForce* [9] 28
Slika 22. Rukavica *P5 data glove* [21] 29
Slika 23. Rukavica *Wearhap* [11] 30
Slika 24. Kontakt s površinskim slojem različite tvrdoće[32] 37
Slika 25. Električni otpor elektrovodljivog tekstila 42
Slika 26. Električni otpor velostata 43
Slika 27. Materijal za izradu senzora, a) elektrovodljiva tkanina, b) velostat, c) pamučna rukavica, d) spužvica 44
Slika 28. Oblikovani slojevi senzora: a) elektroda, b) sloj velostata, c) sloj spužvice, d) pamučna tkanina, e) pomoćni pribor 45
Slika 29. Taktilni senzor osjetljiv na dodir 46

Slika 30. Električni otpori: a) velostata, b) vodljive tkanine 47
Slika 31. Električni otpori senzora: a) neopterećeno stanje, b) predmet postavljen na senzor, c) maksimalan pritisak prstom.....48
Slika 32. Pločica Arduino uno i *breadboard* 49
Slika 33. Shematski prikaz strujnog kruga i Arduino Uno 50
Slika 34. Arduino Uno i spojevi senzora 51
Slika 35. Programski kôd za snimanje odziva senzora.....52
Slika 36. Odnos vrijednosti napona i analognih vrijednosti [22].....53
Slika 37. Prikaz zapisa vrijednosti varijabli za svaki senzor u više ciklusa 54
Slika 38. Grafički prikaz odziva senzora.....55
Slika 39. Gornja strana rukavice prije i poslije stavljanja senzora.....57
Slika 40. Donja strana ruakvice prije i poslije stavljanja senzora.....57
Slika 41. Kabel za rukavicu 58
Slika 42. Shematski prikaz spoja rukavice u strujni krug.....59
Slika 43. Podatkovna rukavica spojena na pločicu Arduino Uno izrađenim kabelom 60
Slika 44. Dijagram toka za uspostavljanje komunikacije 62
Slika 45. Programski kôd u IDE za definiranje komunikacije.....63
Slika 46. Programski kôd za pozivanje funkcije za uspostavljanje komunikacije u MATLAB-u.....63
Slika 47. Programski kôd funkcije za uspostavljanje komunikacije u MATLAB-u.....64
Slika 48. Dijagram toka za očitavanje i ispis vrijednosti senzora.....65
Slika 49. Programski kôd u programu IDE za očitavanje vrijednosti senzora po okidaču iz MATLAB-a 66
Slika 50. Programski kôd u MATLAB-u za pozivanje funkcije za očitavanje stanja senzora 66
Slika 51. Programski kôd funkcije u MATLAB-u za slanje okidača i očitavanje stanja senzora.....66
Slika 52 Programski kôd pozivanja MATLAB-ove funkcije *polyfit* 68
Slika 53 Programski kôd pozivanja MATLAB-ove funkcije *polyval* 68
Slika 54 Programski kôd u MATLAB-u za provođenje kalibracije 1.....68
Slika 55 Programski kôd u MATLAB-u za provođenje kalibracije 2.....69
Slika 56. Aproksimacija na temelju izmjerenih vrijednosti i poznatih masa na kažiprst 71
Slika 57. Flomaster na vagi.....73
Slika 58. Izvedba pokreta GA 74
Slika 59. Snimljene vrijednosti za pokret GA.....75

Slika 60. Pokret GB: a) posezanje, b) hvatanje, c) ispuštanje 76
Slika 61. Snimljene vrijednosti za pokret GB 77
Slika 62. *Breadboard* na vagi 79
Slika 63. Pokret GC (prvi dio): a) posezanje, b) hvatanje 80
Slika 64. Pokret GC (drugi dio): c) držanje, d) ispuštanje 81
Slika 65. Snimljene vrijednosti za pokret GC 82
Slika 66. Snimljene vrijednosti za jedan klik na mišu 85
Slika 67. Snimljene vrijednosti za dvoklik na mišu 86
Slika 68. Uvrtanje i odvrtanje čepa boce 87
Slika 69. Odvrtanje zategnutog čepa boce 88
Slika 70. Uvrtanje i zatezanje čepa boce 89
Slika 71. Držanje USB *sticka* 90
Slika 72. Snimljene vrijednosti za umetanje USB sticka 91
Slika 73. Snimljene vrijednosti za vađenje USB *sticka* 92
Slika 74. Prazna čaša vode na vagi 93
Slika 75. Držanje čaše 94
Slika 76. Snimljene vrijednosti za držanje prazne čaše vode 95
Slika 77. Čaša ispunjena vodom na vagi 96
Slika 78. Snimljene vrijednosti za čašu ispunjenu vodom 97
Slika 79. Tuljac na vagi 98
Slika 80. Držanje tuljca 99
Slika 81. Snimljene vrijednosti senzora za tuljac 100
Slika 82. Kućište parfema na vagi 101
Slika 83. Držanje kućišta parfema 102
Slika 84. Snimljene vrijednosti senzora za kućište parfema 103

POPIS TABLICA

Tablica 1. Odnosi između TMU-a i drugih mjernih jedinica vremena 8
Tablica 2. Postupak postavljanja matice na vijak 19
Tablica 3. Usporedba rezultata ponavljanih mjerenja kažiprsta 70
Tablica 4. Prosječne vrijednosti normalnih sila za snimljene pokrete 83
Tablica 5. Prosječne snimljene vrijednosti za različite predmete 104

SAŽETAK

Promatranjem načina izvođenja tjelesnih pokreta u raznim djelatnostima – proizvodnji, sportu, medicini ili svakodnevnom životu – uviđa se nedostatak informacija o iznosima sila koje su pritom prisutne. U nastojanju za mjerenjem vrijednosti sila koje se pojavljuje pri pokretima šake, razvijena je podatkovna rukavica. Rukavica sadrži tekstilne i taktilne senzore, smještene na svakom prstu. Uz razvijenu programsku podršku, podatkovnom je rukavicom provedeno nekoliko eksperimenata u cilju stjecanja slike o iznosima sila pri izvedbi nekih osnovnih pokreta šakom. Na temelju eksperimenata izneseni su zaključci o važnostima snimljenih vrijednosti te naznačene mogućnosti daljnjeg razvoja podatkovne rukavice.

KLJUČNE RIJEČI: podatkovna rukavica, sila, pozicija, MTM-2, Arduino Uno, tekstilni senzor, taktilni senzor

SUMMARY

By observing ways of performing physical movements in various activities of production, sport, medicine or everyday life, the lack of information on the amount of force that is present is noticed. With this project, in trying to measure the force values that occur in hand movements, data glove have been developed. The glove contain textile and tactile sensors, located on each finger. Along with developed program support, several experiments have been performed with the data glove to gain image of the amount of force in performing some basic movements. Based on the experiments, conclusions were drawn about the importance of the recorded values and the potential for further development of the data gloves.

KEY WORDS: data glove, force, position, MTM-2, Arduino Uno, textile sensor, tactile sensor

1. UVOD

Ljudi su tjelesna bića: od svoga postanka, razvili su brojne vještine koje ostvaruju pokretima svoga tijela. Bilo da se radi o samim počecima – kada čovjek tek otkriva načine preživljavanja te razvija vještine lova, izrade alata, gradnje domova, pa sve do danas – kada postoje načini optimiranog izvođenja pokreta kojima se realiziraju sasvim određene radnje u proizvodnji, medicini, sportu, umjetnosti, zabavi i brojnim drugim područjima: ljudsko je tijelo čovjekov najvrjedniji alat koji koristi svakodnevno. Od trenutka kada čovjek ustane iz kreveta pa sve do kraja dana, izvede množinu pokreta tijelom, više ili manje svjesnih, kako bi mu život tekao na željen način. Čovjekov dan tako se sastoji od niza pokreta koji su već toliko uvježbani i ponavljani da ih se radi nesvjesno. Naprimjer, običnu radnju ispijanja vode iz čaše, najčešće izvodimo a da pritom uopće ni ne razmišljamo o tome što radimo, odnosno, rijetko kad se pritom razmišlja o tome da se čaša mora najprije uhvatiti te prinijeti ustima i naginjati u skladu s razinom (količinom) njenog sadržaja, istodobno s gutanjem u ustima ulivene tekućine. Također, rijetko se kada čovjek pita kolikom silom treba uhvatiti čašu, a da se ona ne razbije. Ljudski mozak i ljudska osjetila ovakve radnje izvršavaju toliko dobro da ne moramo o njima uopće ni razmišljati. Drugi primjer može biti pisanje: pišući o nečemu, autor najčešće razmišlja o sadržaju koji prenosi na papir, a ne o načinu micanja pisala da bi ispisao željena slova, ili potrebnom pritisku pisala o podlogu da ostaje pisani trag, a da se pritom ne oštete pisalo i podloga. Međutim, pisanje je trebalo naučiti, nužno je bilo da se najprije definira kombinacija pokreta kojima se određeno slovo ispisuje, a kasnije se pisanje pretvara u rutinsku radnju. Za potrebe učenja robota da piše, potrebne su precizne informacije o potrebnim silama kojima se drži pisalo i pritišće o podlogu.



Slika 1. Pravilno bacanje loptice u *baseballu* [1]

Postoje mnogo složeniji pokreti, naprimjer u: proizvodnji, sportu i medicini, koji nisu jednostavni, i čiji način izvođenja ima velik utjecaj na rezultat. Naprimjer, u bilo kojem sportu, mnogo toga ovisi o pokretima i o tome tko ih vještije odnosno preciznije izvodi. Sportske pokrete je potrebno opisati kako bi se njihovo izvođenje potom moglo učiti i trenirati (Slika 1.). U industriji, gdje su vrijeme i kvaliteta odlučujući faktori, stalno se teži ka usavršavanju određenih postupaka kako bi posao bio obavljen brže i kvalitetnije. U skladu s tim, postoji i potreba za definiranjem njima pripadajućih pokreta. Međutim, postavljaju se pitanja: kako znati da je određeni postupak bolji ili lošiji od nekog drugog, kako neki pokret uopće opisati nekome, čovjeku ili robotu, koga želimo podučiti određenoj vještini? Sve ovo se svodi na to, da su za opisivanje određenih pokreta potrebne informacije. Naime pokrete je potrebno opisati, a kako bi se moglo prosuditi njihovu kvalitetu, potrebno je izmjeriti određene veličine kojima će se istaknuti njihove karakteristike.

2. STUDIJ POKRETA

Studij pokreta bavi se proučavanjem pokreta koji se koriste u cilju obavljanja neke svrhovite radnje te definiranju optimalnog načina njihova izvođenja. Cilj studija pokreta jest: 1. izbacivanje svakog lošeg ili neučinkovitog pokreta, pokreta kojima se ne postiže korist, a troše se vrijeme i energija, 2. definiranje sheme gibanja kojom se postiže najviša učinkovitost. Studij pokreta može se definirati kao sustavno snimanje i kritičko ispitivanje postojećih i predloženih načina (metoda) rada kao sredstvo za razvoj i primjenu lakše i učinkovitije metode (rada) i time sniženje troškova. [2] Ovo se prvenstveno odnosi na industriju čije stalno rastuće potrebe približavanja optimalnom načinu rada i aktualiziraju ovu znanstvenu disciplinu. Ipak, znanost o pokretima je vrlo bitna i za druga područja u kojima se također može primijeniti. Naprimjer u medicini, osim važnosti shvaćanja načina kretanja ljudskog tijela za potrebne liječenja i rehabilitacije, znanost o pokretima može definirati odmjerene i precizne pokrete, poput onih koji se primjenjuju u kirurškim zahvatima. Sport je također grana u kojoj mnogo toga ovisi o načinu izvođenja određene kretnje koja se može proučiti te definirati optimalan način izvođenja. Studije o kretnjama razvijaju normiranje pokreta, stabilizaciju radnih mjesta obuhvaćajući tako također razvoj i primjenu standardne opreme u standardnim uvjetima te konačno stvaranje osnove za mehanizaciju i automatizaciju rada.

Važne značajke koje definiraju ciljeve studija pokreta jesu [2]:

- osigurati minimalnu opasnost za ljudsko zdravlje,
- pronaći optimalan način izvođenja posla,
- minimiziranje neučinkovitih pokreta,
- efektivno rukovanje materijalom i strojevima,
- kreiranje uputa pojedincima o standardnom načinu izvođenja pojedinih kretnji.

Studij pokreta sadrži nekoliko koraka kojima se pristupa pri opisivanju i poboljšavanju kretnji kod obavljanja nekog rada:

1. korak: Rastavljanje složene operacije na pojedine, jednostavnije radnje

Prvi korak se odnosi na raščlanjivanje zadatka na niz pojedinih radnji čijim se pravilnim redoslijedom postiže željeni cilj. Ovo se vrlo često odnosi na rukovanje pojedinim alatima ili materijalom u procesu obavljanja promatranog posla.

2. korak: Detaljno opisivanje pojedinih operacija

Za detaljan opis načina izvođenja pojedinih operacija te rukovanja odgovarajućom opremom potrebno je proći kroz sljedeća pitanja:

- Cilj
  - Koji je cilj ove operacije?
  - Ispunjava li ova operacija postavljene zahtjeve?
  - Može li se izbjeći?
- Mjesto
  - Gdje je najbolje izvoditi promatranu operaciju?
- Vrijeme
  - Koje vrijeme obavljanja operacije je najprihvatljivije u ergonomskom i ekonomičnom smislu?
- Osoba koja obavlja operaciju
  - Kome dodijeliti pojedinu operaciju?
  - Tko bi ju mogao obavljati brže ili kvalitetnije?

3. korak: Razvoj nove metode

Prije definiranja novog načina obavljanja određenog posla potrebno je proći kroz sljedeće činjenice:

- Eliminacija – za svaku operaciju je potrebno razmotriti posljedice njena izbacivanja iz postupka te donijeti zaključak da li je ona potrebna ili ne.
- Kombinacija – potrebno je razmotriti mogu li se dvije operacije spojiti u jednu i postiže li se time ušteda vremena.
- Reorganizacija slijeda operacija – razmatranje najefektivnijeg slijeda obavljanja operacija.

- Pojednostavnjenje – ako je moguće, odnosno, ako ne stvara negativne utjecaje, potrebno je usvojiti svako pojednostavnjenje izvođenja operacije. Naprimjer: primjena određenih alata i opreme, upotreba obje ruke, odlaganje materijala i alata na odgovarajuća mjesta itd.

4. korak: Uvođenje nove metode u primjenu

Radnici moraju biti trenirani za primjenu nove metode te je rad potrebno promatrati i dodatno unapređivati sve do postizanja zadovoljenja radnika

5. korak: Održavanje metode

Radnici moraju dobiti obrasce za posao kojima se vrše planirane provjere da se utvrdi što se radi u suprotnosti od definirane metode. Radnici moraju biti educirani o izvođenju pokreta za realizaciju pojedinih operacija unutar nove metode.

6. korak: Snimanje podataka metode

Određeni, bitni činitelji metode pokreta kao što su: vrijeme izvođenja, broj neuspješnih pokušaja i slično, bilježe se i prikazuju u obliku grafa.

3. SUSTAVI UNAPRIJED ODREĐENIH VREMENA

Proučavanjem rada (studij rada i vremena) razvijeni su sustavi unaprijed određenih vremena. Sustavi unaprijed određenih vremena su postupci kojima se analizira i raščlanjuje svaka ručna operacija ili metoda rada na osnovne pokrete potrebne za njeno izvođenje, a svakom osnovnom pokretu određuje se jedan unaprijed određeni standard vremena na osnovi prirode pokreta i uvjeta pod kojima se pokret izvodi. [3] Ovim metodama dobivene su mnoge smjernice za oblikovanje metoda rada i radnih mjesta.

Za razvijanje sustava unaprijed određenih vremena najzaslužniji su Taylor[1] i Gilbreth[2]. Gilbreth je u proučavanju studija o radu došao do spoznaje da vrijeme izvođenja određene operacije, uz postavku da ju izvode osobe iste razine uvježbanosti, podjednake spretnosti, sposobnosti te zalaganja, ovisi jedino o metodi rada koja se primjenjuje. Snimanjem pokreta i proučavanjem snimljenih uzoraka primjećuje nekoliko osnovnih pokreta od kojih se može sastaviti gotovo svaka operacija. Ove osnovne pokrete Gilbreth naziva *therblig*. Prva definicija *therbliga* sastojala se od 17 osnovnih elemenata te su pomoću njih razvijene metode rada s najkraćim vremenom njihova izvođenja. Ušteda na vremenu postignuta je eliminiranjem svih nepotrebnih i suvišnih pokreta te primjenom njihovih standardiziranih pokreta odnosno *therbliga*. Uštedom vremena izvođenja operacija postignuto je unaprjeđenje postojećih metoda rada.

Ipak, veliki nedostatak ove metode leži u tome što ovim pokretima nisu dodijeljena unaprijed utvrđena vremena čime bi se pojedini pokreti mogli uspoređivati i kombinirati najefektivnija kombinacija pokreta. Daljnjim proučavanjem studije rada razvija se i sustav unaprijed određenih vremena, točnije, jedan od Gilbrethovih suradnika, Segur objavljuje u periodu od

[1] F. W. Taylor (1856.–1915.) – američki inženjer poznat po radovima na unapređenju industrijske efikasnosti

[2] F. B. Gilbreth (1868.–1924.) – američki inženjer, jedan od začetnika studija pokreta

1919. do 1924. godine prvi sustav unaprijed određenih vremena MTA (*Motion Time Analysis*).

Sustavi unaprijed određenih vremena služe za: definiranje metode rada, utvrđivanje vremenskih vrijednosti za definirane metode rada, poboljšanje postojećih metoda rada, izradu podloga za normiranje (vremenski standard), oblikovanje radnih mjesta, podučavanje radnika u radu. [3]

3.1. Sustav MTM i njegov razvoj

MTM je skraćenica od *Methods – Time Measurement* što se može prevesti kao: Metode – Mjerenje vremena. MTM je postupak, pomoću kojeg se raščlanjuje svaka ručna operacija na osnovne pokrete. Svakom osnovnom pokretu se pridružuje jedna normirana vremenska vrijednost, čiji se iznos određuje na osnovu uvjeta pod kojima se pokret vrši. [2]

Sustav MTM kreirali su stručnjaci na području studija pokreta, H. B. Maynard, J. L. Schwab i G. J. Stegmerten u Sjedinjenim Američkim Državama. Oni postavljaju temelje sustava MTM koji je kasnije dorađivan i usavršavan. Osnovni postupak MTM izlazi 1948. godine u časopisu „Factory Managment and Maintenance“, a iste godine izdana je i knjiga „Methods Time Measurement“. Kasnije se iz osnovnog postupka MTM razvijaju i složeniji nastavci ove metode te nastaju unaprijeđene i dotjerane inačice sustava MTM: MTM-2, MTM-3 i druge. Sustav MTM i danas se intenzivno koristi i stalno razvija i dotjeruje, uključujući i njegove softverske implementacije. [2]

Neki od ovih softvera jesu:

–TICON – Omogućuje brzu i praktičnu analizu korištenja vremena, održavanje i ponovno korištenje formula i izračuna.

– TICON TAKT – Ovo je dodatni modul baziran na TICON softveru kojim je omogućena jednostavna izmjena radnog sadržaja *drag&drop* metodom dok softver automatski po promjeni radi novi ažurirani izračun vremena.

– PROKONDIGITAL – Pomaže u prepoznavanju vremenskih nedostataka uzrokovanih montažom proizvoda te upućuje na alternativne dizajnerske izvedbe kako bi se uštedilo vrijeme u proizvodnji i povećala produktivnost.

Postoje i drugi softveri koji ovdje nisu navedeni te još dodatnih inačica koje se baziraju na softveru TICON. Također, oblikovanje i analiza rada i vremena prisutni su i u softverima CATIA i Siemens NX.

Za definiranje vremena pojedinih kretnji u svrhu razvijanja sustava MTM korišteni su bili video zapisi raznih operacija i zahvata unutar industrijske proizvodnje. Brzina snimanja bila je 16 sličica u sekundi, a brojenjem sličica omogućeno je određivanje vremena za svaki pokret. Promatralo se više sudionika (radnika) te je time stvoreno i određeno rasipanje vremena koja su potom svedena na normalna vremena pomoću postupka normaliziranja tzv. LMS. LMS je skraćenica izvedena od početnih slova imena autora ovog postupka: Lowry, Maynard, Stegmerten. Postupak je poznat još pod nazivom *Levelling*. Normaliziranjem je je definiran normalni učinak koji trajno ostvaruje radnik prosječnih fizičkih i psihičkih sposobnosti, prosječne uvježbanosti i spretnosti bez ikakvih štetnih posljedica. [2]

Za potrebe definiranja vremena trajanja nekog pokreta uvedena je mjerna jedinica TMU koja opisuje vremensku vrijednost i koristi se unutar MTM sustava. TMU je skraćenica koja dolazi od *Time Measurement Unit* i iznosi 0,036 sekundi.

Tablica 1. Odnosi između TMU-a i drugih mjernih jedinica vremena

TMU	SEKUNDE [s]	MINUTE [min]	SATI [h]
1	0,036	0,0006	0,00001
27,8	1	0,017	0,00028
1666,7	60	1	0,017
100 000	3600	60	1

MTM se kasnije brzo proširio i na ostale kontinente, a naročito je izražen interes za usvajanjem MTM sustava u Europi gdje se kasnije također razvijaju nove nadogradnje na postojeći sustav MTM. Primjena MTM-a u industrijskim djelatnostima najizraženija je u SAD-u, Švedskoj, Velikoj Britaniji, Njemačkoj i Švicarskoj. [2]

Značaj postupka MTM je u tome da se umjesto snižavanja troškova naknadnim pojednostavnjenjem rada i oblikovanjem radnog procesa, sprječava uopće nastajanje takvih troškova prethodnim planiranjem metode rada i radnih tokova. [2]

3.2. Sustav MTM-1

Sustav MTM-1 namijenjen je za analizu rada u masovnoj i velikoserijskoj proizvodnji. Ovakvu proizvodnju karakterizira podjela rada na kratke operacije i zahvate uz visok broj ponavljanja, a zahtjevi za točnošću zbog višestrukog ponavljanja ovih operacija su vrlo visoko postavljeni.

Pri nastajanju MTM-a ustanovljeno je da 85 % metoda rada sastavljeno od pet osnovnih pokreta rukama: posezanje, hvatanje, prenošenje, sastavljanje i ispuštanje.[3] Osim osnovnih pet postoji još i tri pokreta koja služe za opisivanje slijeda osnovnih pokreta, a to su pritiskivanje, rastavljanje i zakretanje. Osim pokreta rukama MTM-om su ustanovljena dva osnovna pokreta očima: prebacivanje pogleda i ispitivanje očima (pregledavanje). Zatim je još definirano devet osnovnih pokreta tijelom. Ovih devet grupirano je u tri skupine [3]:

1) Bez pomicanja osi tijela:
    - Pokret stopala
    - Pokret noge
2) S pomicanjem ili zakretom osi tijela:
    - Korak u stranu
    - Zakretanje tijela
    - Hodanje
3) Sa savijanjem stopala
    - Prigibanje i uspravljanje
    - Saginjanje i uspravljanje
    - Klečanje i uspravljanje
    - Sjedenje i ustajanje.

U nastavku ovog rada bit će detaljnije opisani osnovni pokreti rukama.

Posezanje – simbol R (eng. *Reach*)

Ovo je osnovni pokret koji se izvodi u nastojanju da se dovede šaka ili prsti do nekog predmeta ili mjesta.

Najvažnije veličine koje utječu na vrijeme kod ovog pokreta su slučaj posezanja i dužina pomaka šake ili prstiju.

Slučaj posezanja odnosi se na iziskivanje misaonih i vizualnih kontrola pri izvođenju pokreta. Razlikuju se slučajevi kada se poseže rukom prema jednom predmetu i kada se poseže prema jednom predmetu koji je pomiješan u gomili s brojnim drugim predmetima. Posezanja jednake duljine u različitim uvjetima rezultiraju i različitim vremenima izvođenja.

Hvatanje – simbol G (eng. *Grasp*)

Ovaj pokret izvodi se prstima ili šakom u svrhu ostvarenja kontrole nad određenim predmetom.

Faktori koji utječu na vremenske vrijednosti ovog pokreta su položaj predmeta, svojstva predmeta i vrsta hvatanja.

Vrste hvatanja su: hvatanje stiskanjem prstiju, ponovno hvatanje, hvatanje preuzimanjem, hvatanje izabiranjem i hvatanje dodirom.

Prenošenje – simbol M (eng. *Move*)

Ovaj pokret odnosi se na premještanje određenog predmeta premještanjem šake ili prstiju iz jednog mjesta na drugi. Vrlo često, za vrijeme prenošenja, mora biti aktivan i pokret hvatanja.

Najvažnije utjecajne veličine na iznose vremena pokreta su: slučaj izvođenja pokreta, duljina pokreta i masa ili otpor.

Slučajevi prenošenja mogu biti: prenošenje predmeta prema drugoj ruci ili do graničnika, prenošenje predmeta u jedan približni položaj ili prenošenje u točno definirani položaj.

Sastavljanje – simbol P (eng. *Position*)

Sastavljanje je pokret kojim se postiže određeni položaj jednog predmeta u odnosu na drugi. Pokret se izvodi prstima i šakama te zahtijeva visoku razinu kontrole pokreta. Može biti mehaničko i površinsko, površinsko je slučaj namještanja jednog predmeta na drugi.

Faktori koji utječu na vremenske vrijednosti su vrsta spoja i vrsta rukovanja. Vrsta spoja ovisi o dimenzijama i točnosti predmeta kojeg sastavljamo ili točnosti namještanja kod površinskog sastavljanja.

Ako je kod pokreta sastavljanja potrebno izvoditi i dodatne pokrete riječ je o otežanom rukovanju.

Ispuštanje – simbol RL (eng. *Release*)

Ispuštanje je suprotni pokret od hvatanja, kod ispuštanja prestaje kontrola prstiju ili šake nad predmetom.

Razlikuju se dva slučaja pokreta: ispuštanje otvaranjem prstiju, ispuštanje prekidom kontakta.

Pritiskivanje – simbol AP (eng. *Apply Pressure*)

Kod ovog pokreta se ne ostvaruje nikakav pomak predmeta, ostvaruje se samo određena sila u mišićima kako bi se svladao neki otpor.

Zakretanje – simbol T (eng. *Turn*)

Ovo je zakret prazne ili pune šake oko osi podlaktice.

Na vrijeme utječu faktori: kut zakretanja i težina predmeta ili otpor.

Rastavljanje – simbol D (eng. *Disengage*)

Rastavljanjem se za razliku od sastavljanja ostvaruje prekid veze između dva predmeta, tj. postojeći otpor nestaje.

3.3. Sustav MTM-2

Sustav MTM-2 sastavljen je od kombinacija osnovnih pokreta definiranih u okviru MTM-1 sustava. Sastoji se od ukupno 11 elemenata. Ovaj sustav idealan je za analizu ručnih standardnih operacija u serijskoj proizvodnji, može se koristiti i za poduku radnika o obavljanju određenih operacija. MTM-2 nastao je od strane švedskog MTM udruženja kao težnja za stvaranjem univerzalno primjenjivog MTM sustava s najčešće ponavljanim

sljedovima osnovnih pokreta. Do takve ideje dovodi rezultat istraživanja o učestalosti pojave osnovnih pokreta i kombinacije pokreta.

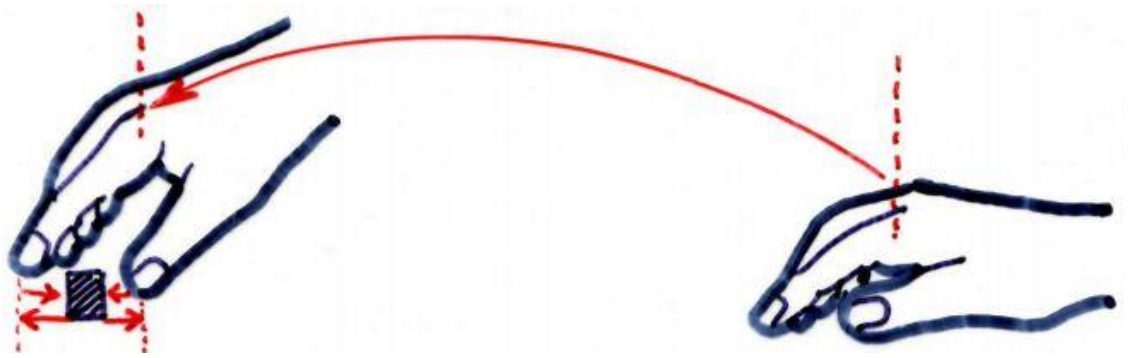
MTM-2 sustav sveden je na sljedeće pokrete: uzimanje, dodatak za težinu kod uzimanja, postavljanje, dodatak za težinu kod postavljanja, ponovno hvatanje, pokreti očiju, okretanje ručice, pokret stopala, korak, prigibanje i uspravljanje. U nastavku će biti opisani neki pokreti koji se odnose na djelovanje rukama.

Uzimanje – simbol G (eng. *Get*)

Uzimanje je slijed pokreta čiji je cilj posezanje do nekog/ih predmeta, zatim hvatanje jednog ili više predmeta te kasnije ispuštanje predmeta (Slika 2.). Veličine koje utječu na vrijednosti vremena su slučajevi uzimanja, dužina pokreta i težina predmeta ili otpor.

Slijed osnovnih pokreta:

UZIMANJE = POSEZANJE + HVATANJE + ISPUŠTANJE.

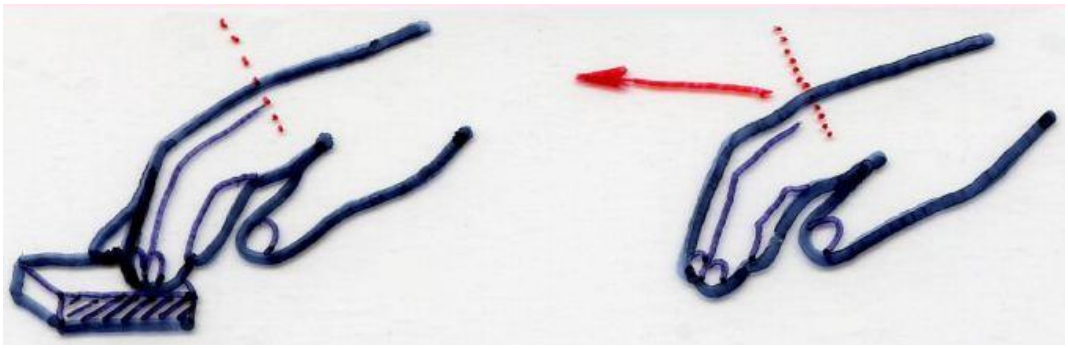


Slika 2. Pokret uzimanje G [4]

Tri su slučaja uzimanja, kako slijedi.

Slučaj GA

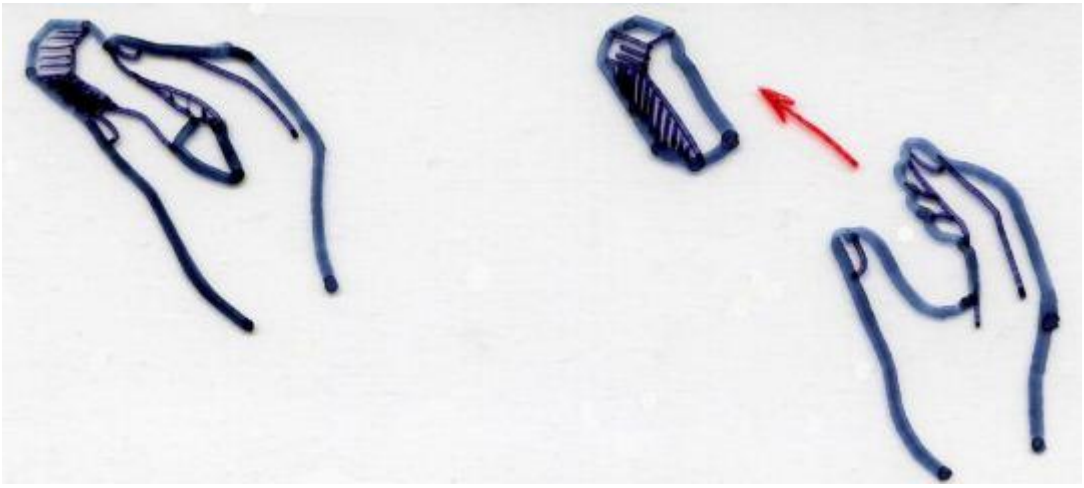
Ovo je slučaj uzimanja koji se odlikuje time da se kontrola nad predmetom ostvaruje bez pokreta hvatanja već dodirom (Slika 3.).



Slika 3. Slučaj uzimanja GA [4]

Slučaj GB

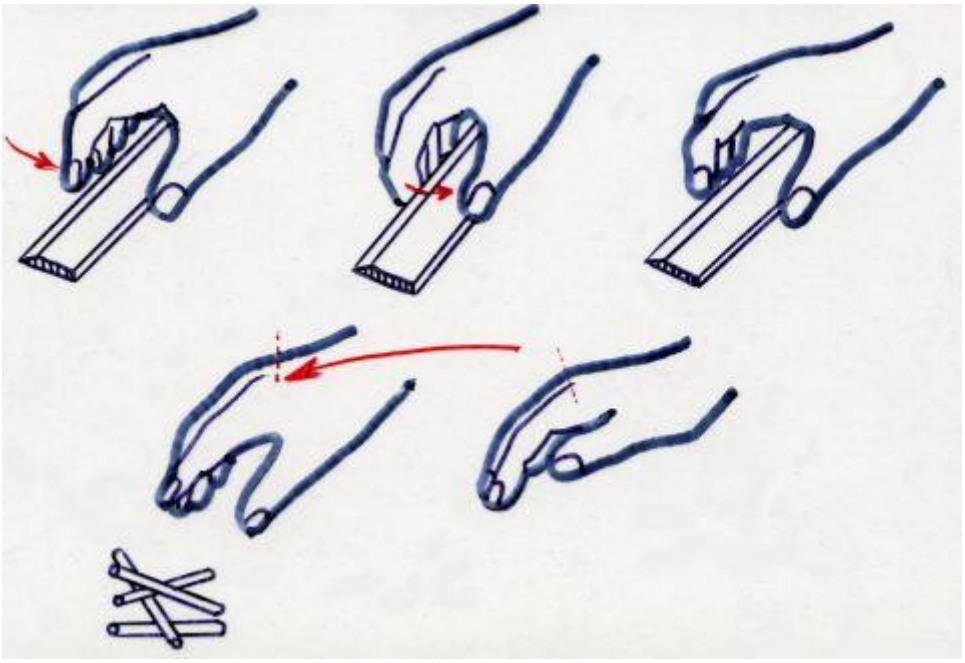
Odnosi se na uzimanje nekog predmeta prikladnog oblika, jednostavnog za hvatanje (Slika 4.). Ovo je slučaj kad je potreban samo jedan pokret hvatanja jednostavnim zatvaranjem šake ili stiskanjem prstiju.



Slika 4. Slučaj uzimanja GB [4]

Slučaj GC

Kod ovog slučaja potrebno je više pokreta hvatanja da bi se ostvarila određena kontrola nad željenim predmetom (Slika 5.). Npr., kod plosnatih, tankih predmeta koji leže na ravnoj podlozi potrebno je jedan kraj predmeta malo izdići kako bi se omogućilo kontrolirano hvatanje te uzimanje predmeta.



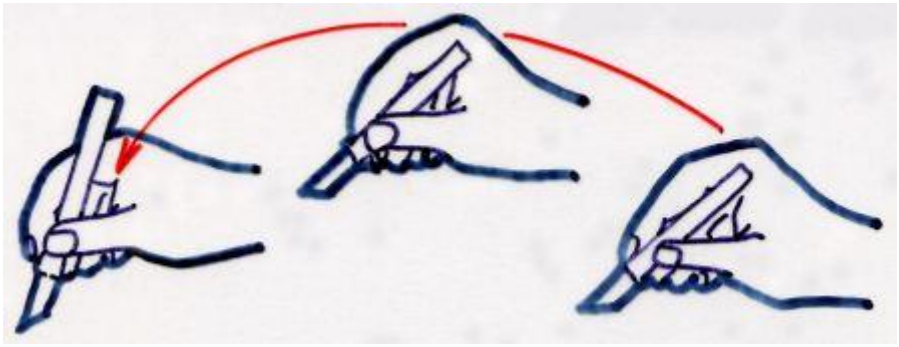
Slika 5. Slučaj uzimanja GC [4]

Postavljanje – simbol P (eng. *Put*)

Ovo je sustav pokreta koji ima za cilj da se neki predmet ili skupina predmeta premjeste iz jednog položaja i/ili mjesta na neki drugi položaj i/ili mjesto (Slika 6.). Na vrijednosti vremena utječu faktori: slučaj postavljanja, dužina pokreta, masa ili otpor.

Postavljanje se realizira na sljedeći način:

POSTAVLJANJE= PRENOŠENJE + SASTAVLJANJE (ili samo prenošenje).

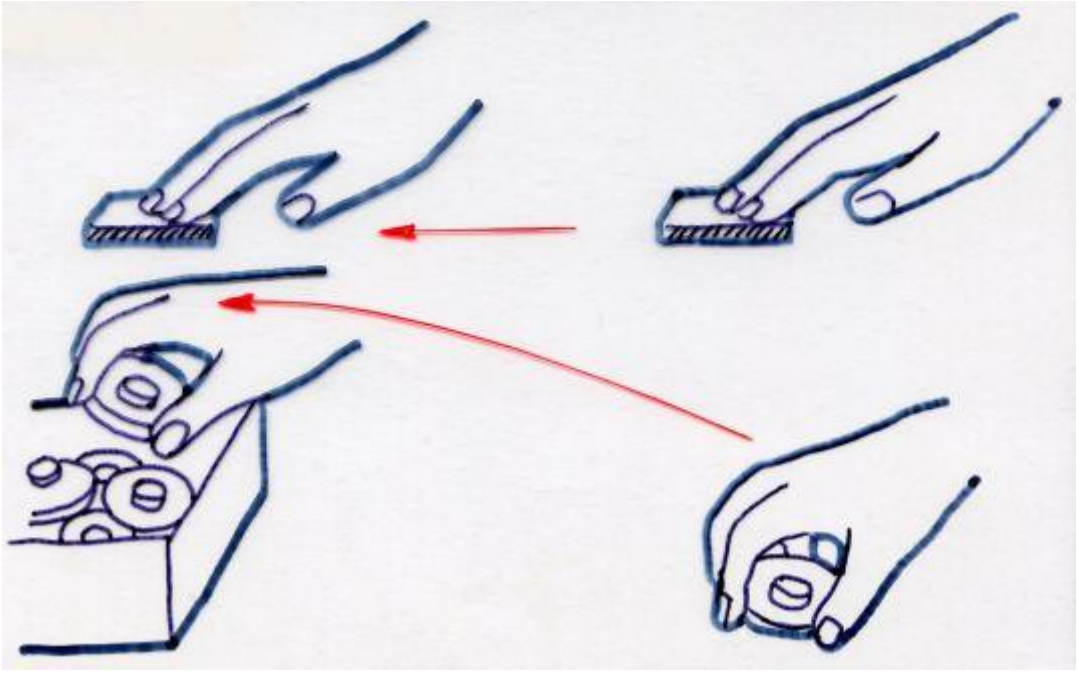


Slika 6. Pokret postavljanje P [4]

Slučajevi postavljanja razlikuju se prema broju pomoćnih, korektivnih pokreta.

Slučaj PA

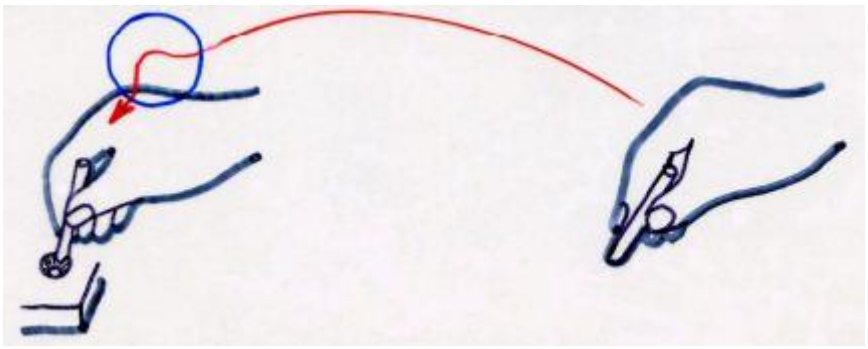
Ovo je slučaj bez korektivnog pokreta i vrlo se često koristi (Slika 7.).



Slika 7. Slučaj postavljanja PA [4]

Slučaj PB

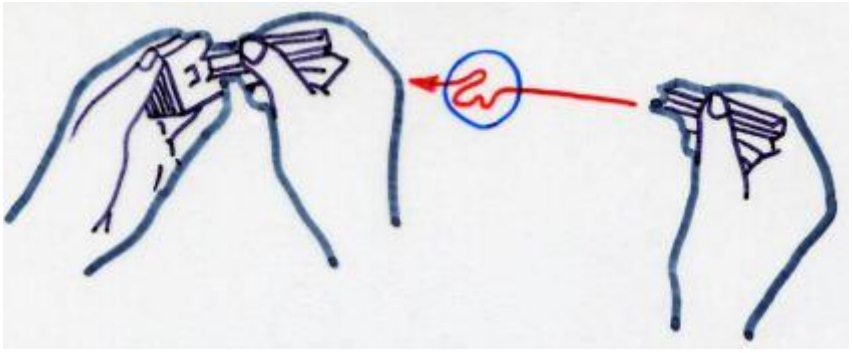
Slučaj postavljanja PB zahtjeva jedan korektivni pokret, pojavljuje se najčešće kod sastavljanja dvaju predmeta s labavim dosjedom (Slika 8.).



Slika 8. Slučaj postavljana PB [4]

Slučaj PC

Za ovaj slučaj potrebna su dva ili više korektivnih pokreta, najčešće su to kratki, nenamjerni, teško predvidivi pokreti koji imaju za posljedicu poteškoće kod rukovanja (Slika 9.). Ovi korektivni pokreti pojavljuju se zbog prijelaznih dosjeda, nesimetričnosti dijelova, neadekvatnog položaja pri radu i slično.



Slika 9. Slučaj postavljanja PC [4]

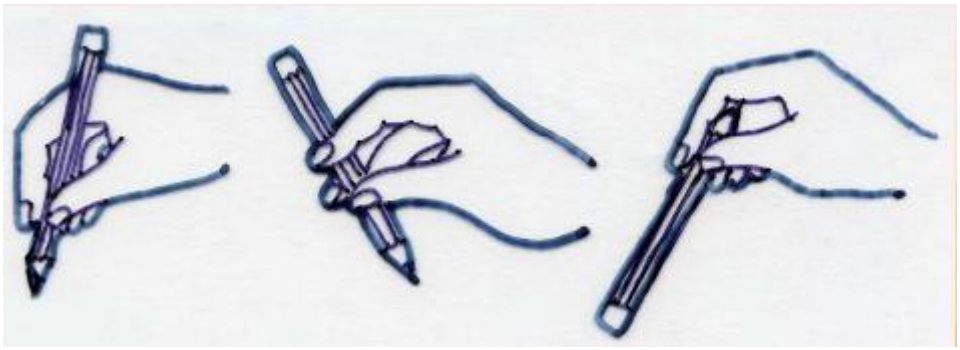
Dodatak za masu kod uzimanja i postavljanja GW i PW

Dodatak za masu kod uzimanja obuhvaća aktivnosti pripreme mišića šake i prstiju za svladavanje sile gravitacije u ovisnosti o masi predmeta. Ovo je statička komponenta i uglavnom se broji 1 TMU za svaki kilogram mase. Pojavljuje se nakon što su šaka ili prsti obuhvatili predmet u pokretu uzimanja, prsti moraju biti zatvoreni prije nego se izvede bilo koji drugi pokret.

Dodatak za masu kod postavljanja ovisi o masi predmeta koji se prenosi. Dodatkom za masu obuhvaća se dodatno vrijeme zbog manje brzine kod prenošenja težih predmeta. Uglavnom se dodaje 1 TMU na 5 kg mase predmeta. Dodatak započinje pokretom prenošenja i završava poslije završnog pokreta prenošenja.

Ponovno hvatanje – simbol R (eng. *Regrasp*)

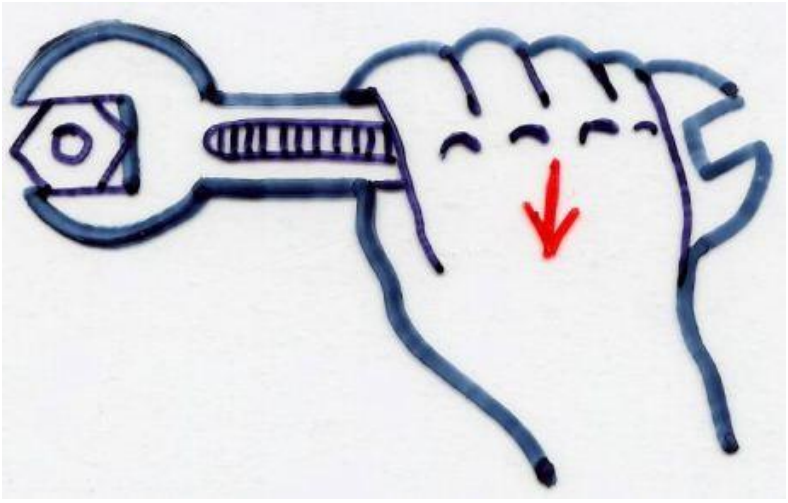
Ponovno hvatanje je slijed kratkih pokreta s ciljem da se uspostavi bolja kontrola nad nekim predmetom odnosno uspostavi bolja sigurnost od ne ispadanja predmeta (Slika 10.). Sastoji se od maksimalno tri mala pokreta prstima.



Slika 10. Pokret ponovno hvatanje R [4]

Pritiskivanje – simbol A (eng. *Apply pressure*)

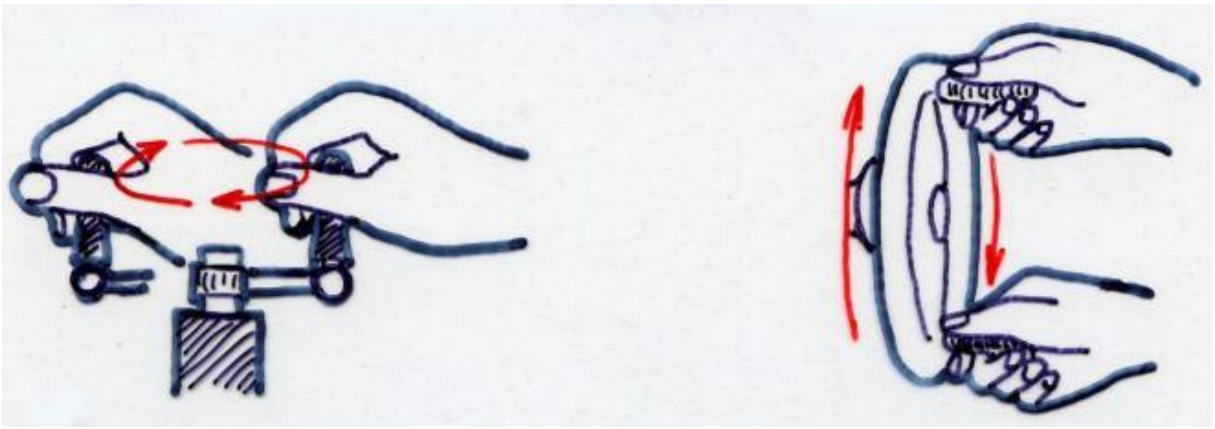
Pritiskivanje je primjena snage mišića. Ovaj pokret počinje nakon što je određeni dio tijela u dodiru s predmetom te djeluje kontrolirani porast snage mišića, određeno zadržavanje sile i na posljetku otpuštanje sile (Slika 11.). Pokret završava nakon otpuštanja sile, a promatrani dio tijela je još uvijek u dodiru s predmetom.



Slika 11. Pokret pritiskivanje A [4]

Okretanje ručice – simbol C (eng. *Crank*)

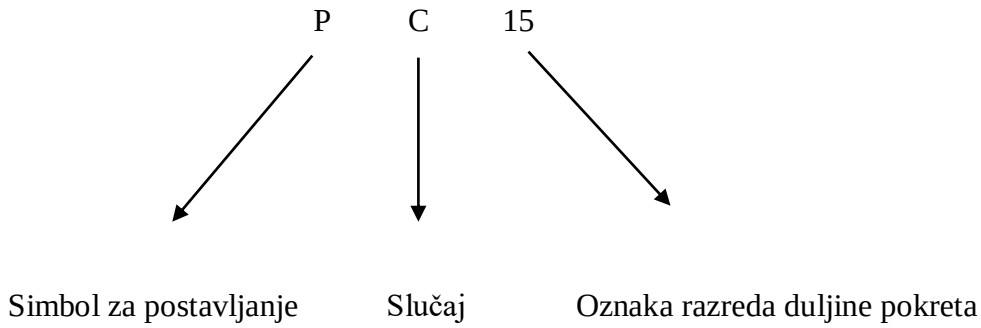
Ovo je pokret kojim se šakom ili prstima predmet premješta po kružnoj putanji (Slika 12.). Utjecajne veličine na iznos vremena su broj okreta i masa ili otpor.



Slika 12. Pokret okretanje ručice C [4]

3.4. Označavanje i način zapisivanja pokreta u sustavu MTM-2

Svaki pokret kao dio slijeda pokreta neke promatrane radnje zapisuje se navođenjem simbola samog pokreta. Nakon simbola pokreta navodi se slučaj, a iza slučaja razred duljine pokreta – Slika 13.



Slika 13. Primjer zapisivanja pokreta u sustavu MTM-2

Slijed svih kretnji zapisuje se ovim načinom označavanja slijedno unutar tablice s pripadajućim iznosima vremena (TMU).

Naprimjer, postupak uzimanja matice iz spremnika (30 cm) i postavljanje na vijak (30 cm) te uvrtanje matice s 10 pokreta prstima i pritezanjem prstima zapisuje se na način prikazan tablicom 2.

Tablica 2. Postupak postavljanja matice na vijak

TMU	SIMBOL	OPIS RADA DESNE RUKE
23	GC30	Maticu (uzeti) iz spremnika
30	PC30	Postaviti (maticu) na vijak
6	2PA5	Tražiti početak navoja, dva puta
70	10GB5	Obuhvatiti 10 puta
30	10PA5	Uvrtati 10 puta
14	A	Pritegnuti maticu (prstima)

3.5. Sile pri pokretima

Značaj MTM-a je u tome što su identificirani elementarni pokreti koji su jasno opisani te se kombinacijama ovih pokreta može izvesti i opisati gotovo svaki tjelesni rad. Do izražaja sustav MTM dolazi u industriji, ali pogodnosti ideje MTM-a mogu se iskoristiti u svim djelatnostima, i to: ne samo za bolje planiranje, nego i za kvalitetno i precizno obavljanje bilo kakvih operacija u bilo kojim djelatnostima.

U proizvodnji je vrijeme vrlo značajan faktor te se metoda MTM veže upravo oko mjerenja vremena pojedinih pokreta kako bi se stvorio pregled raspolaganja s vremenom. Međutim, postoje i druge bitne veličine koje se vežu uz pokrete. Jedna od takvih veličina je i sila koja se javlja djelovanjem rukama ili prstima na neki predmet. Bitan je značaj primijenjene sile na vrijeme trajanja samog pokreta. Može li se, naprimjer, u uputstvima snažan pritisak točnije definirati ili razmotriti posljedice na objekt ili ljudsku ruku uslijed pritiska? Informacije o silama uslijed obavljanja elementarnih pokreta mogu se iskoristiti i u području virtualne

stvarnosti[3]. Dakle, postoje važni razlozi zašto bi se pokreti sustava MTM-a obogatili informacijama o silama koje se pojavljuju. Upravo je to i motivacija ovoga rada, naime, **istražiti mogućnosti prikupljanja podataka relevantnih za pokrete, konstrukcijom i primjenom podatkovne rukavice, i to financijski nezahtjevne**.

[3] Naime, MTM se sam po sebi može smatrati nekom vrstom virtualne stvarnosti, adekvatne dobu nastanka sustava MTM. MTM je odraz i opis fizičke stvarnosti, a danas su na raspolaganju tehnologije još kvalitetnijeg – dubinskog opisa stvarnosti: uz virtualnu stvarnost, tu je i tzv. *real physics*.

4. PODATKOVNE RUKAVICE

Podatkovne su rukavice (eng. *data gloves*) uređaji u obliku rukavica koje se navlače na šaku i prste, a čovjeku omogućuju nesmetano rukovanje predmetima kao i bez nje. Na sebi imaju smještene jedan ili više senzora kojima se prikupljaju podatci o fizikalnim veličinama pri određenim djelovanjima ruke. To mogu biti naprimjer, kut savijanja prstiju, sile koje djeluju na prste ili šaku, položaj prstiju ili šake, temperatura i ostalo. Uređaj je kabelom ili bežično povezan s računalom gdje određeni softver daje prikaz o prikupljenim podatcima ili se pak prikupljeni podaci koriste za neku određenu svrhu. U većini slučajeva podatkovne rukavice su izlazni (*output*) uređaji, ali postoje slučajevi gdje se koriste i u suprotnu svrhu. To se odnosi na slučajeve gdje podatkovne rukavice mogu sadržavati uređaje koji djeluju na ruku, odnosno čovjeka, simulirajući osjet dodira. Ovakve rukavice primjenjuju se upravo za simuliranje osjeta dodira kod virtualne stvarnosti. U engleskom jeziku podatkovne rukavice poznate su pod nazivima: *data gloves, wired gloves* ili *cybergloves*.

4.1. Podatkovna rukavica Sandinea i Defantija – *The Sayre glove*

Jedan od prvih pokušaja razvijanja podatkovne rukavice dogodio se na Sveučilištu Illinois u Chicagu gdje je razvijena podatkovna rukavica pod nazivom *The Sayre glove* (Slika 14.). Na razvoju ovog uređaja radili su Daniel J. Sandine[4] i Thomas Defanti[5] 1977. godine.

Ovaj uređaj pratio je kretnje šake te pružao mogućnost za višedimenzijsku kontrolu nad skupinom objekata. Rukavica je sadržavala senzore koji su snimali savijanje prstiju. Senzor se

[4] Daniel J. Sandine (rođen 1942. godine) – američki dizajner, istraživač i umjetnik na području video zapisa i računalne grafike

[5] Thomas Defanti (rođen 1948. godine) – američki istraživač na području računalne grafike

sastojao od fleksibilnih cjevčica koje su postavljene uzduž gornje strane prstiju na rukavici. Na jednoj strani cjevčice nalazio se izvor svjetla, a na drugoj fotoćelija osjetljiva na količinu svjetla kojoj je izložena. Uslijed savijanja prstiju, a time i cjevčice, količina registrirane svjetlosti je varirala te je na taj način zabilježeno savijanje prstiju. Rukavica se koristila pri baratanju kliznim upravljačima pri čemu je zabilježeno potrebno savijanje prsta za postizanje određene vrijednosti određene veličine kliznim upravljačima. Ovim izumom se prvi puta pojavljuje ideja o razvijanju sustava za prepoznavanje gestikulacije.



Slika 14. Podatkovna rukavica Sandinea i Defantija – *The Sayre glove* [19]

4.2. Zimmermanova podatkovna rukavica – *The data glove*

Thomas G. Zimmerman[6] 1982. godine razvija patent (Slika 15.) koji je bio sličan prethodnom primjeru *The Sayre glove*. Patent se sastojao od rukavice na kojoj su smješteni optički senzori savijanja kojima je mjereno savijanje prstiju. Optičke senzore savijanja izumio je Young L. Harvil[7] koji je na optičkim vlaknima izgrebao površinu na mjestima blizu zglobova prstiju. Na taj način razlikovao se signal koji putuje optičkom niti za slučaj kada je nit savijena i kada nije. The data glove se smatra kao prva podatkovna rukavica koja je bila komercijalno dostupna. Zimmerman je kasnije nastavio raditi na ugradnji ultrazvučne i magnetske tehnologije za praćenje položaja ruke što je preteča razvoja na području virtualne stvarnosti.



Slika 15. Zimmermanova podatkovna rukavica – *The data glove* [20]

4.3. Rukavica *Nintendo power glove*

[6] Thomas G. Zimmerman – američki istraživač i izumitelj na području interakcije čovjeka i računala

[7] Young L. Harvil – američki istraživač, razvio niz patenata na području uređaja virtualne stvarnosti

Nintendo power glove (Slika 16.) razvijen je kao upravljač za igranje video igara. Igrom se upravljalo pokretima ruke. Rukavica je sadržavala četiri senzora savijanja, za svaki prst osim za mali prst. Pretpostavka je bila da mali prst uglavnom prati pokrete ostalih prstiju te da nije toliko bitan, a izostavljanjem pripadajućeg senzora postigla se ušteda na izradi rukavice. Ovim senzorima dobiveni su podatci o tome koliko je pojedini prst, osim malog prsta savinut, a signalima dobivenim savijanjem prstiju upravljalo se određenim djelom igre. Osim savijanja prstiju rukavica je snimala i poziciju ruke odnosno rukavice. Uređaj obuhvaća i stalak na kojem se nalaze tri ultrazvučna senzora. Stalak se postavlja kraj televizora u položaj da senzori čine vrhove pravokutnog trokuta. Odašiljanjem ultrazvučnih signala s rukavice pomoću dva ultrazvučna odašiljača senzori prepoznaju poziciju rukavice.



Slika 16. Rukavica *Nintendo power glove* [8]

4.4. Rukavica *Cyberglove*

Podatkovnu rukavicu pod imenom *CyberGlove* izvorno je razvila 1990. godine kompanija Virtual Tehnologies[8].

CyberGlove proizvodi su podatkovne rukavice koje prikupljaju informacije o kretnjama korisnikove ruke te bežičnom komunikacijom i odgovarajućim softverom daju prikaz na računalu. Ovime pružaju korisniku mogućnost da manipulira digitalnim predmetima na računalu kao da se radi o fizički stvarnim objektima

CyberGlove podatkovne rukavice danas su razvijene od strane kompanije CyberGlove Systems LLC u četiri osnovne serije. Prva serija je poznata pod imenom samo *CyberGlove*, ovo je osnovna linija rukavica koje su opremljene senzorima kojima se određuje pozicija ruke. Koriste se senzori savijanja koji su postavljani na gornjoj strani prstiju rukavice te se njima detektira savijanje pojedinih prstiju. Osim senzora savijanja postoji još niz drugih senzora kojima se snima pozicija i orijentacija ruke. Postoje dvije inačice unaprijeđene verzije klasične *CyberGlove* rukavice, a to su *CyberGlove II* i *CyberGlove III*. *CyberGlove II* razvijen je u izvedbama s 22 i 18 senzora. Manja izvedba, s 18 senzora (Slika 17.), izvedena je sa slobodnim vrškovima za prste kako bi se olakšali neki postupci, naprimjer tipkanje po tastaturi. Ovaj proizvod sastoji se od senzora savijanja, dva ili tri po svakom prstu, ovisno o izvedbi, senzora rotacije i savijanja na zglobovima prstiju i šake te na ručnom zglobu. Senzori su vrlo tanki i lagani te su potpuno neuočljivi na rukavici. Rukavica prima napajanje preko baterije te podržava bežičnu tehnologiju.

[8] Nju 2000. godine preuzima kompanija Immersion Corporation. CybreGlove liniju proizvoda kasnije, 2009. godine stječe kompanija CyberGlove Systems LLC.



Slika 17. Rukavica *CyberGlove II* [9]

CyberGlove III (Slika 18.) razvijen je na temelju 20 godina iskustva na prijašnjim *CyberGlove* rukavicama. Ova serija CyberGlove rukavica sadrži unaprijeđenu tehnologiju senzora koji daju preciznije, ponovljive i čiste podatke o poziciji ruke. Osim naprednijih senzora sadrži još niz poboljšanja kao što su naprednija bežična komunikacija s računalom, veća količina memorije, mogućnost rada s uštedom energije. ergonomski dizajn, a kompatibilna je i s softverima za prijašnje modele rukavica.



Slika 18. Rukavica *CyberGlove III* [9]

CyberTouch je serija *CyberGlove* (Slika 19.) proizvoda koja je opremljena vibrirajućim stimulatorima na svakom prstu kojima se simulira osjet dodira. Pri rukovanju nekim

predmetom u virtualnoj stvarnosti, ovi stimulatori na pojedinim prstima vibriraju određenim intenzitetom ne bi li stvorili privid o dodirivanju nekog predmeta. Razvijena je također i novija inačica nazvana CyberTouch II. Ova verzija ima lakše vibro stimulatore koji pružaju realističniji osjet dodira od prethodne verzije.



Slika 19. Rukavica *CyberTouch* [9]

CyberGrasp (Slika 20.) model je izveden u obliku egzoskeleta koji dolazi s rukavicom te pruža određeni otpor prstima pri njihovom savijanju. Otpor savijanju proporcionalan je tvrdoći nekog predmeta koji je simuliran u virtualnoj stvarnosti. Ovim načinom stvara se privid o stiskanju nekog čvrstog ili spužvastog predmeta u virtualnoj stvarnosti. Osim stvaranja osjeta o stiskanju nekog virtualnog predmeta, ovaj uređaj daje i informacije o silama koje su simulirane u virtualnoj stvarnosti.



Slika 20. Rukavica *CyberGrasp* [9]

Posljednja, najkompletnija inačica ovog proizvoda dolazi pod imenom *CyberForce*. (Slika 21.). *CyberForce* osim pružanja otpora prema prstima u svrhu simuliranja osjeta stiskanja predmeta, simulira osjet otpora na cijelu šaku. Ovim uređajem moguće je simulirati oslanjanje šake naprimjer na virtualni volan. Također, mjere se i snimaju podatci o virtualnim silama koje djeluju na cijelu šaku.



Slika 21. Rukavica *CyberForce* [9]

4.5. Rukavica *P5 dataglove*

Ovaj model (Slika 22.) razvijen je od tvrtke Essential Reality kao ulazna jedinica za računala kao zamjena za uobičajeno primjenjivane miševe. Ovom rukavicom, položajem ruke, pomiče se kursor na upravljačkom sustavu računala. Rukavica ima tvrdi, plastični dio koji dolazi s gornje strane šake na kojem su smješteni odašiljači infracrvenih svjetlosnih signala. Uređaj obuhvaća infracrveni prijemnik koji na temelju primljenih signala određuje pozicija šake.

Rukavica se također sastoji i od pet senzora savijanja koji su smješteni na gornjoj strani pojedinih prstiju. Položajem ruke upravlja se kursorom na ekranu kao što se to uobičajeno radi putem miša, dok se savijanjem prstiju pozivaju neke funkcije kao što su lijevi i desni klik na mišu. Ova rukavica uglavnom je namijenjena za zabavu, odnosno igranje video igara, ali nije ispunila očekivanja. Operativnim sustavom puno je jednostavnije upravljati putem miša, ruka se primjenom rukavice dosta i umarala, a ni doživljaj igranja video igara nije ispunio očekivanja.



Slika 22. Rukavica *P5 data glove* [21]

4.6. Rukavica *WearHap*

Na sveučilištu Bielefeld u Njemačkoj, Neuroinformatički odjel razvija taktilnu osjetnu rukavicu *WearHap – Wearable Haptics for humans and robots* (Slika 23.). Rukavica je osjetljiva na dodire te ujedno mjeri i daje prikaz o iznosu sile koja djeluje na pojedini dio šake. Rukavica je izvedena na način da na unutarnjem dijelu šake ima smješteno 60 elastičnih tekstilnih senzora osjetljivih na dodir. Ovim uređajem moguće je dobiti sliku o silama koje djeluju na prste i šaku pri rukovanju raznim predmetima.



Slika 23. Rukavica *Wearhap* [11]

4.7. Primjena podatkovnih rukavica

Podatkovne rukavice najčešće su izvedene kao uređaji za snimanje pozicije i orijentacije ruke te se tako koriste za prikupljanje podataka o poziciji i orijentaciji šake te položaju pojedinih prstiju. Ovi podatci se vrlo često koriste na području virtualne stvarnosti te je svaki pokret moguće preslikati u virtualni prostor na računalu, a također se vrlo često koristi za manipuliranje ili upravljanje pojedinim funkcijama na računalu, strojevima ili robotima.

Funkcija kontrole softvera podatkovnim rukavicama vrlo je atraktivna na području video igara gdje je moguće podići razinu realizma pri igranju, a također i razinu zabave. Osim zabave, podatkovne rukavice prikladne su za simuliranje određenih radnji čime se mogu definirati određeni pokreti. Simulacijama unutar virtualne stvarnosti stječe se uvid u prednosti i nedostatke određene metode obavljanja poslova (rada), a zapisuju se i podatci kojima je moguće naučiti robota da obavlja istu radnju. Za velik broj pokreta i radnji, robota je lakše naučiti na način da se taj isti pokret snimi dok ga radi čovjek nego da se ručno upisuju koordinate pojedinih prstiju i šake što može biti vrlo složen postupak čak i za čovjeku jednostavnije radnje.

Osim snimanja pozicije, podatkovne rukavice koriste se za snimanje opterećenja koja se javljaju pri obavljanju pojedinih radnji. Opterećenja koja djeluju na šaku ljudske ruke vrlo su bitan podatak za ergonomsko oblikovanje pojedinih proizvoda i procesa. Također, u industriji, za planiranje određenih proizvodnih procesa, osim samog poznavanja vremena izvođenja

pojedinih radnji, bitan podatak je i opterećenje koje djeluje na šaku i prste. Poznavanjem opterećenja moguće je steći preciznije i možda potpuno nove podatke o vremenima izvođenja određenih kretnji. Podatcima o opterećenju šake unutar nekog vremenskog intervala pri obavljanju određenih radnji dobiva se slika o kvaliteti nekog zahvata. Na ovaj način moguće je definirati bolju ili lošiju metodu nekog zahvata koji se obavlja rukom na temelju podataka o reaktivnim silama koje djeluju na ruku. Ukoliko se robota želi naučiti da radi neku radnju umjesto čovjeka, bitno je poznavati kojim silama uhvatiti neki predmet. Važnost poznavanja iznosa sila koje djeluju na pojedine prste pri određenim radnjama naročito je bitno kada se za obavljanje iste radnje želi naučiti robot humanoidnog oblika s izvedbom ruke nalik čovjekovoj.

Posljednji oblik primjene podatkovnih rukavica je primjena kao uređaja koji simulira dodir ili postojanje nekog objekta. Određenim uređajima poput vibrirajućih stimulatora ili mehanizma koji pružaju otpor prstima moguće je simulirati osjet dodira. Na taj način moguće je raditi s virtualnim predmetima koji su opipljivi. Ovo omogućava razne simulacije rada s predmetima koji se mogu nalaziti i biti vidljivi unutar virtualne stvarnosti. Naprimjer, neki kirurg si ne može dopustiti da uvježbava potreban stisak ili neki drugi zahvat na pravom pacijentu. Podatkovnom rukavicom moguće je provesti simulaciju. Za ovakvu primjenu podatkovnih rukavica potrebno je najprije realizirati snimanje opterećenja. Simuliranje osjeta dodira moguće je ako postoje podatci o silama koje djeluju na prste i šaku pri određenim radnjama.

5. OSJETILA PODATKOVNE RUKAVICE

Podatkovnim rukavicama najčešće se prikupljaju podatci o položaju – poziciji i orijentaciji te opterećenju pojedinih prstiju i šake. S obzirom na to, osjetila ili u nekom radnom prostoru te iznosu opterećenja koja se pojavljuju rukovanjem raznim alatima ili objektima te koja djeluju na samu šaku i pojedine prste. U skladu s vrstom snimanih podataka, senzori podatkovnih rukavica se generalno dijele na senzore za snimanje pozicije šake i prstiju te one (senzore) sila i momenata.

5.1. Senzori pozicije šake

Senzorima pozicije ostvaruje se saznanje o koordinatama šake ili pojedinih prstiju u nekom radnom prostoru. Promatranjem promjene koordinata u vremenu stječe se uvid o putanjama ostvarenim kretnjama šake.

U nastavku je dan pregled tehnologija i izvedbi senzora za prepoznavanje pozicije šake u radnom prostoru.

5.1.1. Senzori na mehaničkoj konstrukciji

Kod ovakvog načina praćenja pozicije šake, podatkovna rukavica sastoji se od dodatnog mehanizma, egzoskeleta, nalik spomenutom primjeru podatkovne rukavice *CyberForce* (Slika 21.). Navedeni mehanizam opremljen je na pojedinim mjestima senzorima kojima mjeri dužinu pomaka ili kut zakreta (potenciometri, enkoderi, rezolveri) pojedinih elemenata konstrukcije. Na temelju izmjerenih vrijednosti određuje se i pozicija same rukavice odnosno šake u prostoru. Ovakvim načinom dobiva se vrlo precizna informacija o poziciji, a prednost

je i niska cijena. Ipak, dodatna konstrukcija može vrlo često ometati čovjeka u radu na način da predstavlja prepreku u radnom prostoru ili čovjeka dodatno umara jer se dio energije utrošene u radu koristi i za pomicanje konstrukcije. Također i ograničen domet mehaničkom konstrukcijom predstavlja nedostatak ovog pristupa.

Senzori koji se koriste na zglobnim mjestima podatkovne rukavice jesu:

– Potenciometri

Potenciometrima se može mjeriti rotacijski ili dužinski pomak na način da se u strujnom krugu nalazi otpornik čiju se vrijednost otpora može mijenjati pomakom klizača. Pomak klizača uvjetovan je pomakom podatkovne rukavice te se mjerenjem električnih veličina može zaključiti i iznos nekog pomaka.

– Enkoderi

Enkoderi su digitalni pretvarači koji detektiranjem impulsa određuju rotacijski ili dužinski pomak. Sastoje se od trake ili diska koji su podijeljeni na segmente koji propuštaju ili ne propuštaju svjetlosni signal na fotoosjetljivu diodu. Mogu biti inkrementalni koji broje impulse ostvarene pomakom te apsolutni kod kojih se za svaku poziciju razlikuje jedinstvena kombinacija impulsa.

– Rezolveri

Rezolveri su rotacijski električni transformatori kod kojih se inducira izmjenični napon konstantne amplitude čiji fazni pomak ovisi o kutu zakreta rotora. Mjerenjem faznog pomaka određuje se i kut zakreta.

5.1.2. Optički senzori

Optičko praćenje pozicije realizira se postavljanjem jednog ili više odašiljača nevidljive infracrvene ili vidljive crvene svjetlosti. Postavljanjem kombinacije od dva ili više prijemnika u radni prostor koji mjere intenzitet odaslane svijetlosti zaključuje se iz koje pozicije dolazi signal te se time određuje pozicija šake. Ovakvi senzori pogodni su za primjenu podatkovne

rukavice kod upravljanje nekim sustavom pokretima ruke. Sustav je osjetljiv na prepreke u radnom prostoru te je teško dobiti informacije o kretanjima ruke kod rukovanja alatima i naročito unutar nekog predmeta. Optičko praćenje pokreta može biti realizirano i postavljanjem markera na pojedinim mjestima rukavice koji se snimaju kamerom. Razvijenim vizijskim sustavom i odgovarajućim algoritmom prate se pomaci pojedinih markera te se time dobiva slika pomaka šake.

5.1.3. Akustički senzori

Sličan princip kao kod optičkih senzora (5.1.2.) je i kod akustičkih senzora. Odašiljačima postavljenim na podatkovnoj rukavici odašilje se zvuk visoke frekvencije (ultrazvuk). Emitirani zvuk u radnom prostoru dolazi do prijemnika – specijalnih mikrofona koji na temelju intenziteta dobivenog ultrazvuka šalju signale kojima se računaju koordinate pozicije. Prednosti ovog sustava su niska cijena dobar domet, ali veliki nedostatak je osjetljivost na smetnje okoline. Između rukavice i prijemnika ne smije biti prepreka, a sustav ne podržava veći broj istovremenih odašiljača.

5.1.4. Elektromagnetski senzori

Ovim pristupom koristi se izvor elektromagnetskog polja čija je pozicija fiksna i poznata. Na rukavici se nalaze senzori koji primaju elektromagnetski signal te se na temelju dobivenih signala unutar elektromagnetskog polja računa pozicija ruke. Ovakvi senzori su cijenom pristupačni, ali osjetljivi su na metalne predmete u okolini.

5.1.5. Kapacitivni senzori

Kapacitivni senzori sastoje se od kondenzatora kod kojih se kapacitivnost kondenzatora mijenja u ovisnosti o poziciji ruke u prostoru kojom se mijenja dielektrična konstanta između elektroda ili zbog promjena razmaka među elektrodama kada se je jedna elektroda nalazi na šaci. Kombinacijom od tri okomito postavljene ploče, može se zaključiti koordinata u 3D radnom prostoru.

5.2. Senzori orijentacije

Za određivanje rotacije šake oko vlastitih osi mogu se iskoristiti optički giroskopi. Optički giroskopi su vrlo mali uređaji koji stanu na dlan ruke te ne sadrže pokretne dijelove. Rade na principu istovremenog odašiljanja dva jednaka snopa svijetlosti kroz optičke niti jednakih duljina. Na kraju svake niti nalazi se senzor koji detektira emitiran snop svijetlosti. Zakretanjem senzora oko pojedinih osi jedan snop svijetlosti će do senzora doći prije od drugog. Na temelju ove vremenske razlike računa se i rotacija oko osi šake.

5.3. Senzori savijanja prstiju

Za indikaciju nalazi li se pojedini prst u ispruženom ili savinutom stanju mogu se iskoristiti fleksijski senzori. Ovi senzori su duguljastog oblika, prigodnog za smještanje na prste podatkovne rukavice. Na gornjoj strani senzora nalaze se vodljive čestice koje se razmiču jedna od druge savijanjem senzora odnosno prsta. Razdvajanjem čestica raste otpor dok njihovim sabijanjem otpor pada. Mjerenjem električnih veličina može se zaključiti u kojoj mjeri je prst savijen. Osim opisanih fleksijskih senzora za detekciju savijanja prstiju može se iskoristiti i princip opisan u točki 4.2. gdje se na temelju refleksije snopa svijetlosti izvodi podatak o mjeri savijanja prsta.

5.4. Senzori sile

U svrhu ostvarivanja osjeta dodira i osjeta iznosa opterećenja na prstima podatkovne rukavice potrebni su senzori koji mogu pokriti površinu djelovanja prstiju, ali su dovoljno tanki kako ne bi svojim oblikom smetali čovjeku u radu. Za te potrebe primjenjuju se taktilni senzori koji predstavljaju umjetnu kožu na podatkovnoj rukavici.

5.5. Taktilni senzori

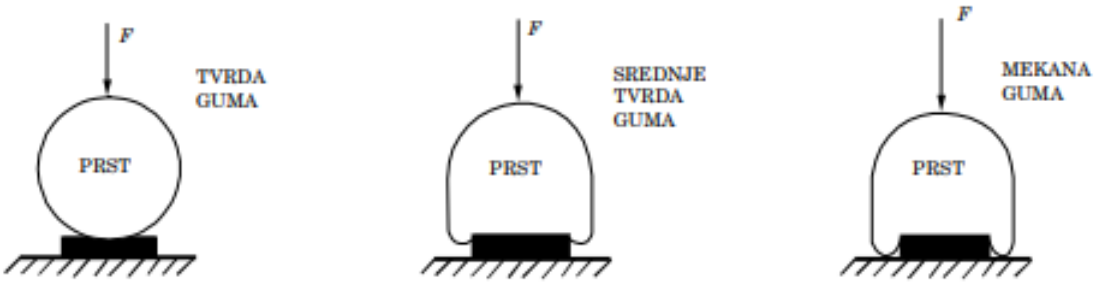
Kod senzora riječ *taktilan* dolazi kao pridjev koji označuje senzor s osjetom opipa odnosno senzor osjetljiv na dodir. Ovim senzorima snima se i mjeri površinski kontakt. Mjere se fizikalne veličine koje daju informacije o okolini kao što su: pozicija i orijentacija nekog objekta, mehanička svojstva objekata (naprimjer: masa, elastičnost, plastičnost), te stanja (naprimjer: vlažnost ili temperatura). Taktilna percepcija definirana je kao kontinuirana detekcija kontaktnih sila na određenoj površini i s određenom prostornom rezolucijom. [32]

Pri hvatanju nekog objekta u najjednostavnijem slučaju s dva prsta, mjesto dodira prsta i objekta može biti točka, linija ili površina. Kontaktna sila je rezultat mase objekta i sile stiska prstiju te se sastoji od normalne komponente okomite na površinu dodirne plohe te tangencijalne komponente paralelne s plohom dodira. Tangencijalne sile rezultat su trenja između dodirnih ploha prsta i objekta te su uzrok smicanju i momentu torzije.

Opća podjela taktilnih senzora je na pasivne i aktivne. Pasivni senzori detektiraju vrijednosti sile i momenta u jednoj točki te se na taj način dobiva informacija o prisutnosti objekta te iznosu reaktivne sile koja djeluje na senzor. Aktivni senzori djeluju duž cijele linije ili na nekoj površini pri čemu mogu odrediti i pozicija ili oblik objekta. Za realizaciju aktivnih taktilnih senzora, pojedini senzori su poredani u niz ili matricu. Prsti šake gibaju se nezavisno te ih je dovoljno opremiti po jednim senzorom na svakom prstu za postizanje aktivnog sustava senzora.

Izlazni signali taktilnih senzora proporcionalni su silama nastalim kod kontaktnog djelovanja te se tim signalima mogu izvesti sljedeći zaključci: prisustvo objekta, oblik, lokacija i orijentacija objekta, intenzitet, smjer i mjesto djelovanja sila, statička i dinamička svojstva kontaktne površine.

Struktura taktilnih senzora sastoji se od tri osnovna dijela: dodirnog sloja, senzora poremećaja i elektronskog sklopa. Funkcija dodirnog sloja (umjetna koža) sastoji se u pretvaranju vanjske sile u mehanički poremećaj te da štiti konstrukciju od vanjskog utjecaja. Ovisno o tvrdoći dodirnog sloja postižu se različite kontaktne površine te se razvijaju različita kontaktna naprezanja. Ove dvije veličine su međusobno obrnuto proporcionalne, tvrđim slojem postiže se mala kontaktna površina, ali se razvija veće kontaktno naprezanje. S mekanim površinskim slojem postižu se suprotni rezultati, veća kontaktna površina uz manje kontaktno naprezanje.



Slika 24. Kontakt s površinskim slojem različite tvrdoće[32]

Senzori poremećaja pretvaraju određenu deformaciju dodirnog sloja u električni signal. Postoji više izvedbi senzora namijenjenih u ovu svrhu, a osnovne karakteristike svih takvih senzora su minijaturnost i prostorna rezolucija. Elektronski sklop omogućava obradu signala dobivenih od senzora. Putem analogno digitalnih pretvornika mikroprocesor prima signale senzora u digitalnom obliku koji daju informacije o promatranim veličinama te se snimljeni signali mogu interpretirati, naprimjer, grafički na računalu.

Obrada taktilne slike slična je procesiranju optičke slike. Kao što je slika sastavljena od piksela, tako se i osjetna površina sastoji od niza taktela koji mogu biti poredani u liniju ili na površini duž više linija.

5.5.1. Otpornički taktilni senzori

Kod otporničkih taktilnih senzora koriste se promjenjivi otpornici čija se vrijednost otpora mijenja s pritiskom na njih, odnosno uslijed djelovanja opterećenja na površinu senzora. Ovakvim načinom dobivaju se različiti signali ovisno o intenzitetu dodira površine senzora te je moguće dobiti sliku o nivou opterećenja senzora. Otpornost ovakvih senzora u neopterećenom stanju iznosi 1 do30 kΩ dok minimalan otpor u opterećenom stanju iznosi 5 do 100 Ω. Kao otpornici s promjenjivim otporom uslijed dodira koriste se najčešće vodljivi elastomeri, a postoji više konstrukcijskih izvedbi otporničkih taktilnih senzora. Otpornički dio može biti postavljan na pločicu s tiskanim kontaktima, može se nalaziti između elektroda ili između nizova elektroda poredanih u linije itd. Otporničke taktilne senzore moguće je izraditi u raznim oblicima i dimenzijama što ih čini pogodnim za primjenu na podatkovnim rukavicama.

5.5.2. Kapacitivni taktilni senzori

Kapacitvni taktilni senzori rade na principu promjene kapaciteta kondenzatora uslijed dodira na senzor. Senzor je zapravo kondenzator koji se sastoji od dviju elektroda između kojih se nalazi izolator. Pritiskom na senzor mijenja se razmak između elektroda, a time i električne vrijednosti senzora. Dobiveni podatci u ovisnosti o intenzitetu pritiska pretvaraju se u iznos opterećenja.

5.5.3. Piezoelektrični taktilni senzori

Izvedba piezoelektričnih senzora vrlo je slična otporničkim. (5.5.1.) Sastoji se od dvije razmaknute elektrode koje su međusobno na tijelu senzora povezane putem piezoosjetljivog materijala. Djelovanjem sile na piezoelektrični materijal nastaje deformacija u strukturi materijala koja dovodi do polarizacije molekulskih dipola. [12] Između elektroda pojavljuje

se naboj koji je proporcionalan sili opterećenja. Signal koji daje piezoelektrični senzor je napon kojim se detektira dodir i intenzivnost dodira ili sila kojom se djeluje na senzor.

5.5.4. Ultrazvučni taktilni senzori

Princip rada ultrazvučnih taktilnih senzora bazira se na mjerenju vremena potrebnog da se emitirani ultrazvuk vrati, odnosno prođe put od odašiljača ultrazvuka, objekta te natrag do prijemnika. Sastoji se od jedne zajedničke, vrlo tanke elektrode na kojoj je sloj pijezopolimera. Sloj pijezopolimera je debljine 30 μm te se lako oblikuje prema prstima šake. S gornje strane pijezopolimera nalaze se pojedinačne elektrode, a preko njih sloj elastične gume do 3 mm. Dovođenjem električnog signala generira se ultrazvuk te vrijeme puta ultrazvuka ovisi i tome u kojoj mjeri je gumeni dio sabijen.

5.5.5. Optički taktilni senzori

Optički taktilni senzori baziraju se na principu prekidanja ili refleksije optičkih zraka. Realizirani su nizovima optičkih vlakna duž površine senzora, a postavljaju se u redove i stupce te tvore matricu parova optičkih vlakana. Pritiskom na pojedinim mjestima sabijaju se optička vlakna te dolazi do prekidanja ili reflektiranja svjetlosti te se dobiva signal o pritisku na određenom mjestu površine senzora.

6. RAZVOJ VLASTITE PODATKOVNE RUKAVICE

U svrhu dobivanja informacija o opterećenjima koja djeluju na prste ljudske šake pri rukovanju alatom ili nekim drugim objektom cilj je razviti sustav koji će prikupljati takve informacije i ujedno ih interpretirati. Navedeni sustav bit će realiziran rukavicom osjetljivom na dodir koja na temelju određenih fizičkih podražaja šalje podatke (signale) računalu. Računalo tada iz prikupljenih signala u realnom vremenu interpretira dobivene vrijednosti u obliku iznosa sile koja djeluje na pojedini prst.

U pregledu sustava MTM uočen je nedostatak podataka o iznosima opterećenja pri definiranim pokretima. Vrijeme je bitan faktor u proizvodnji, međutim, pri planiranju vremena potrebno je voditi i računa o tome da je riječ o ljudskim bićima. Ljudska bića se međusobno razlikuju te ne mogu svi obavljati iste poslove istom brzinom s istim iznosima opterećenja. Također, vrijeme trajanja nekog posla razlikuje se u ovisnosti o iznosu opterećenja, naprimjer, čovjeku treba više vremena da podigne tešku kutiju od one koja je lagana.

Nadogradnjom sustava MTM s izmjerenim prosječnim vrijednostima opterećenja za svaku skupinu pokreta, posao bi se mogao kvalitetnije planirati. Otvorila bi se mogućnost za grupiranje određene vrste poslova prema adekvatnim skupinama ljudi, a i sami pokreti, alati i strojevi mogu biti oblikovani da čovjeka što manje umaraju pri radu, sve u cilju postizanja veće efikasnosti.

Određivanjem sila na pojedinim prstima pri hvatanju nekoga predmeta (ili alata) kojim bi se obavila neka radnja, saznale bi se vrlo korisne informacije za razvoj robota odnosno njegove prihvatnice, za obavljanje iste radnje automatski. Također, i za potrebe simuliranja osjeta dodira kod čovjeka, potrebne su informacije o iznosima sila pri rukovanju određenim objektima.

Većina postojećih podatkovnih rukavica orijentirana je na prikupljanje podataka o poziciji prstiju i šake: u ovome je pak radu namjera **razviti podatkovnu rukavicu s osjetom dodira,**

tj. razviti podatkovnu rukavicu za prikupljanje podataka o silama koje nastaju pri izvođenju pokreta.

6.1. Taktilni senzori vlastite podatkovne rukavice

U svrhu ovog rada razvijeni su tekstilni taktilni senzori kojima se realizira osjetljivost na intenzivnost dodira što se šalje računalu te će kasnije biti interpretirano kao iznos opterećenja. Senzori su sastavljeni od dvije elektrode između kojih se nalazi materijal promjenjivog otpora, a priključuju se u istosmjerni strujni krug. U neopterećenom stanju otpor takvog materijala bit će maksimalan, a s pritiskanjem senzora otpor će se početi smanjivati u ovisnosti o intenzivnosti pritiska. Mjerenjem promjene električnih veličina u strujnom krugu u ovisnosti o promjeni intenziteta pritiska na senzor dobivaju se informacije o opterećenju senzora.

U ovu svrhu odabrani su tekstilni senzori jer se zbog svog sastava lako mogu izraditi u bilo kojem obliku. To su elektrovodljiv tekstil i velostat. Lako ih je oblikovati u oblik jagodice prsta kako bi senzori što bolje prianjali uz sami prst rukavice. Za oblikovanje tekstilnih senzora osim nadalje navedenih materijala dovoljne su škare ili neki drugi oblik rezala.

Elektrovodljiv tekstil

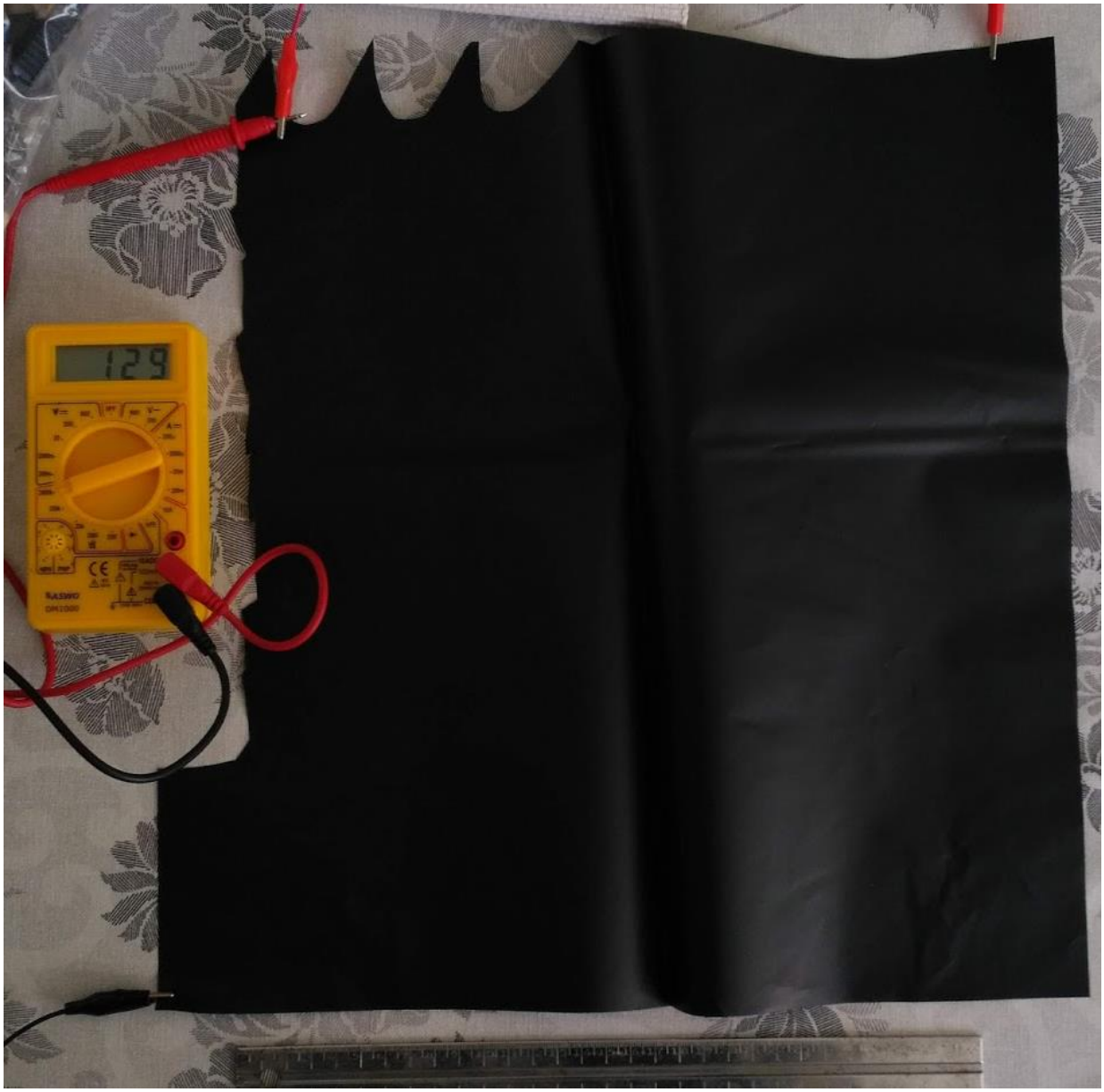
Elektrovodljiv tekstil je materijal srebrne boje koji provodi električnu struju. Elektrode su iz ovog materijala ručno izrezane u prilagođeni oblik za primjenu na prstima ruke. Ovaj materijal sastoji se od bakrom ili niklom metaliziranih niti upletenih u konstrukciju tekstila što daje odličnu električnu vodljivost. Mjerenjem u bilo kojem smjeru i do 30 cm izmjereni otpor ovog materijala iznosi svega 6 Ω (Slika 25.). Na slici 25. se radi zornosti dimenzija uz elektrovodljivi tekstil nalazi i ravnalo.



Slika 25. Električni otpor elektrovodljivog tekstila

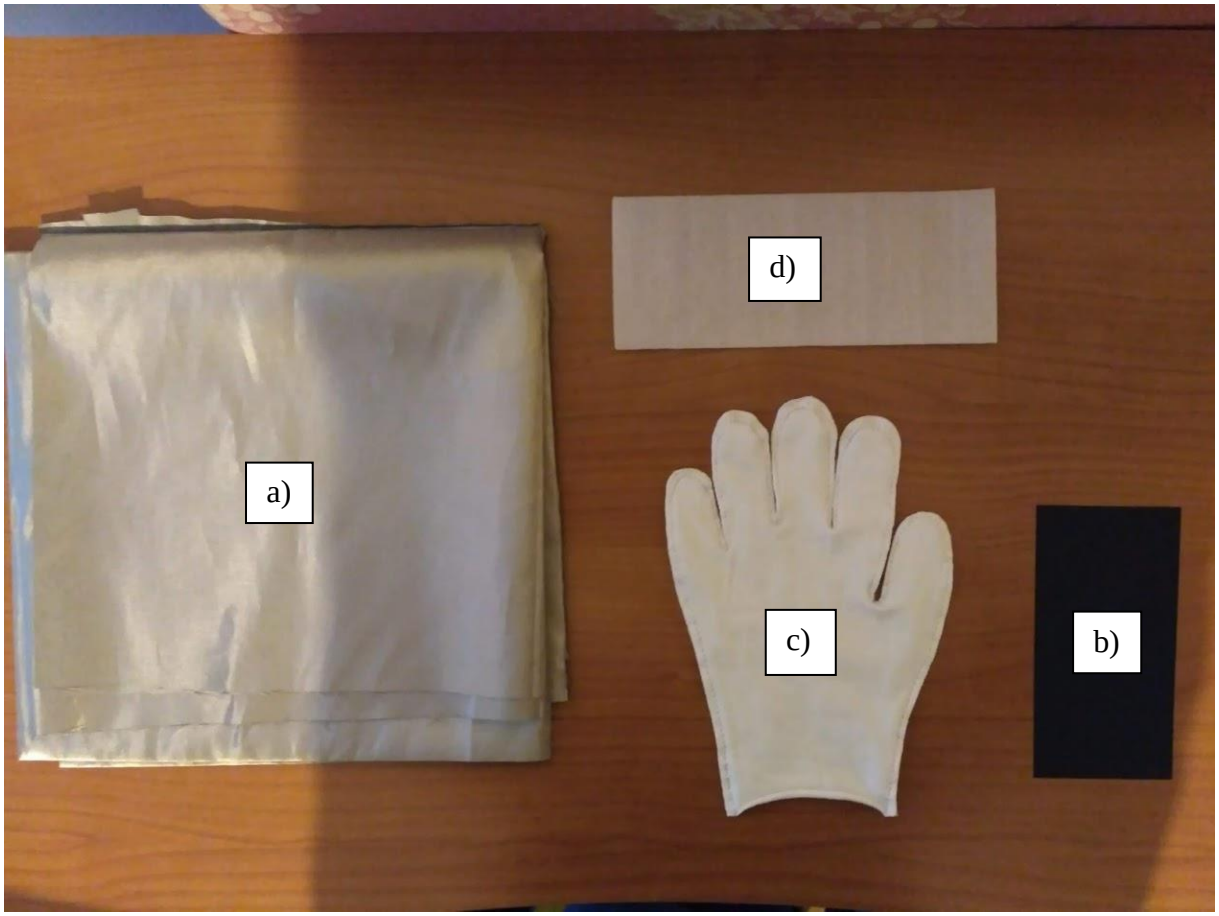
Velostat

Kao posrednik promjenjivog otpora između elektrodi korišten je velostat. Velostat je materijal crne boje koji pritiskom mijenja električna svojstva, odnosno mijenja električni otpor. Sastoji se od polimerne folije impregnirane ugljičnim vlaknima što ovom materijalu daje električnu provodnost, međutim, otpor ovog materijala mijenja se pod djelovanjem kontaktnog opterećenja. Izmjerena vrijednost električnog otpora komada velostata u bilo kojem smjeru duljine i do 30 cm te neopterećenom stanju, iznosi oko 130 kΩ (Slika 26.). Na slici 26. se radi zornosti dimenzija uz elektrovodljivi tekstil nalazi i ravnalo.



Slika 26. Električni otpor velostata

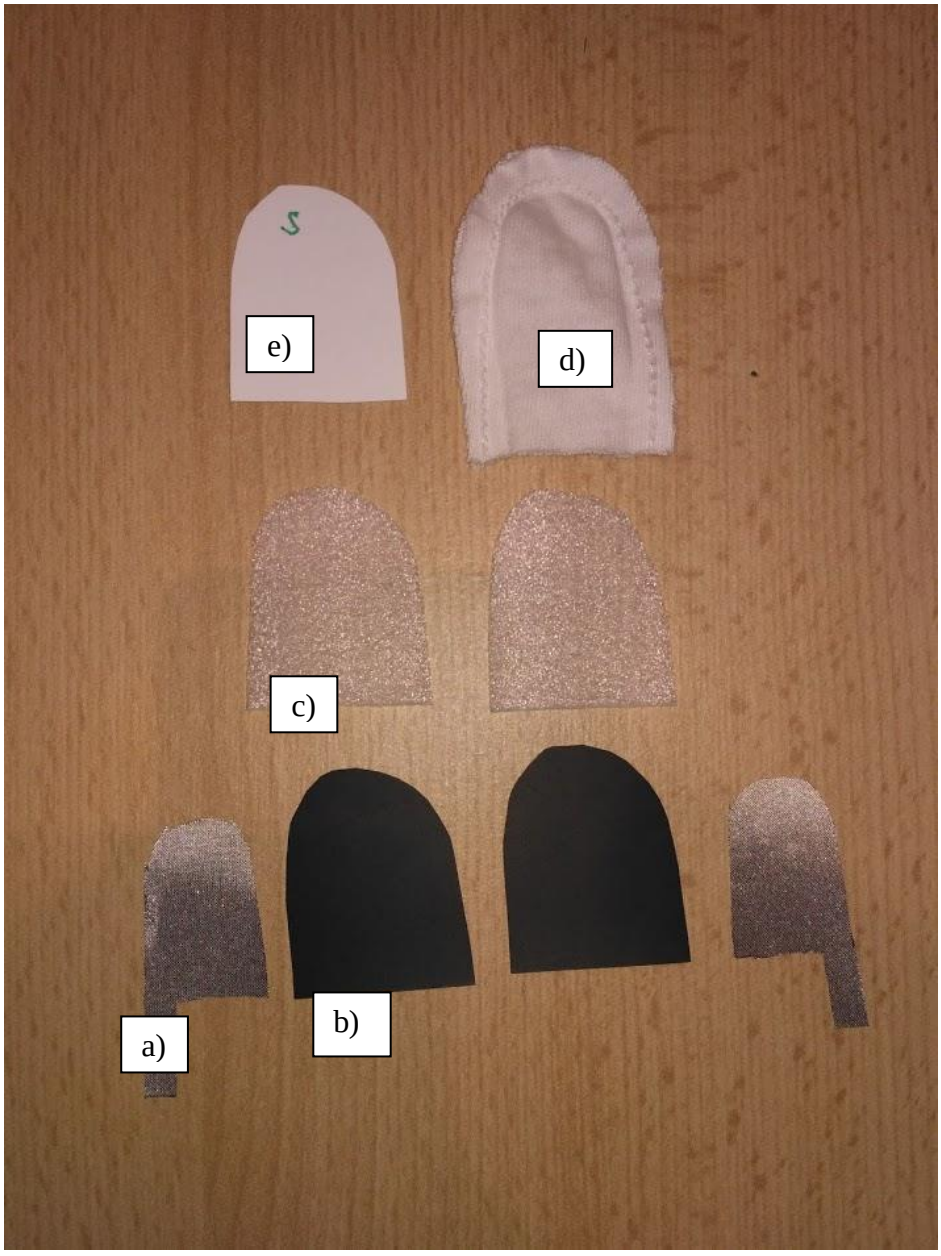
Osim elektrovodljivog materijala (tkanine), za izradu senzora korištena je i pamučna rukavica za lakše oblikovanje senzora prema prstima šake te spužvica za ublažavanje dodira (Slika 27.).



Slika 27. Materijal za izradu senzora, a) elektrovodljiva tkanina, b) velostat, c) pamučna rukavica, d) spužvica

Elektrode senzora realizirane su rezanjem vodljive tkanine u prikladan oblik za primjenu na prstima ruke. Na sličan način izrezani su i slojevi velostata koji se u senzoru umeću između elektrodi. Slojevi od velostata sličnih su kontura kao kod elektrodi, ali većih površina kako ne bi došlo do izravnog dodira pojedinih elektroda. Spužvasti dio istog je oblika i površine kao i sloj velostata. Realizacija senzora ostvarena je nizanjem slojeva materijala prema sljedećem rasporedu:

1) sloj spužvice,
2) elektroda,
3) 2 sloja velostata,
4) elektroda,
5) spužvica.



Slika 28. Oblikovani slojevi senzora: a) elektroda, b) sloj velostata, c) sloj spužvice, d) pamučna tkanina, e) pomoćni pribor

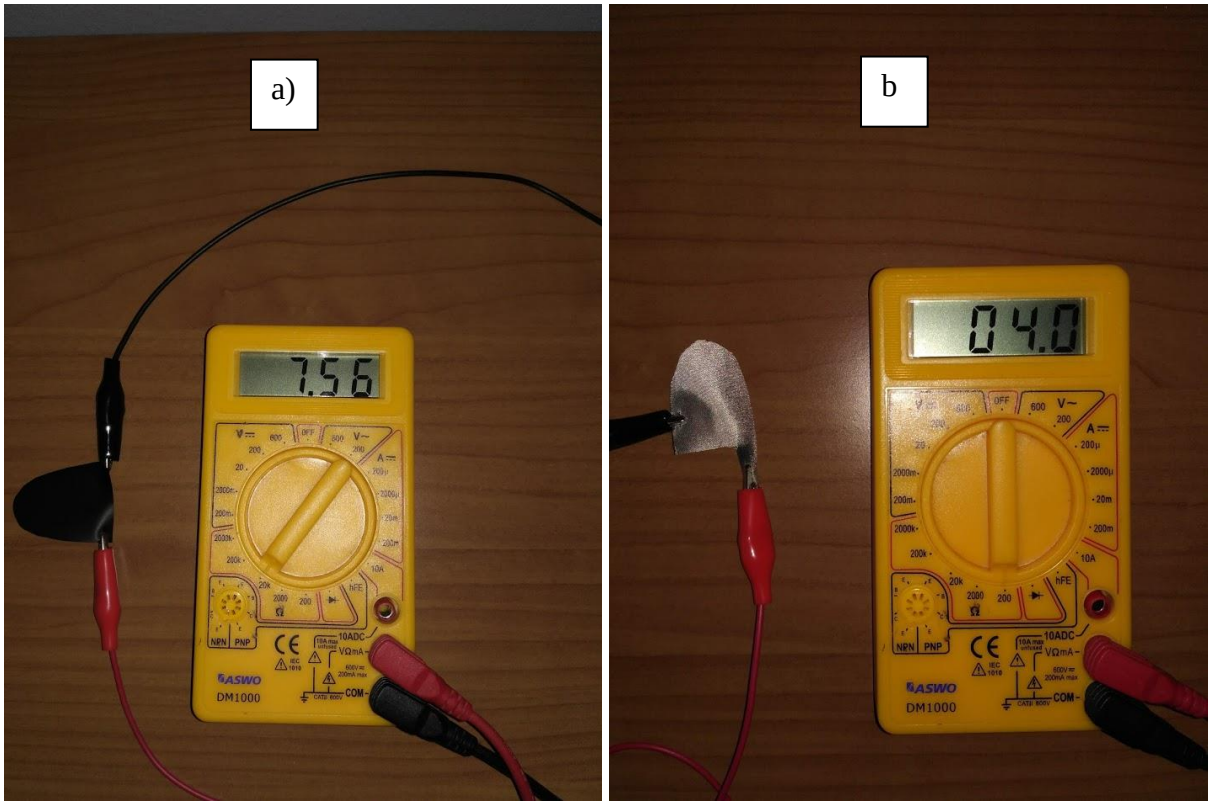
Svi ovi slojevi nanizani jedan na drugi prema navedenom rasporedu smještaju se i zatvaraju unutar pamučne tkanine kako bi bili fiksirani i zatvoreni od okoliša.



Slika 29. Taktilni senzor osjetljiv na dodir

6.1.1. Električna svojstva pojedinih slojeva senzora

Elektrode od vodljive tkanine ispitivanjem su pokazale odličnu električnu vodljivost uz izmjereni električni otpor u prosjeku od 4 Ω, izmjerenog u bilo kojem smjeru. Velostatu u neopterećenom stanju otpor u prosjeku iznosi oko 8 kΩ a pritiskom se taj otpor smanjuje i do nekoliko stotina ohma (Slika 30.). Spužvasti materijal i pamučna tkanina ne provode električnu struju.

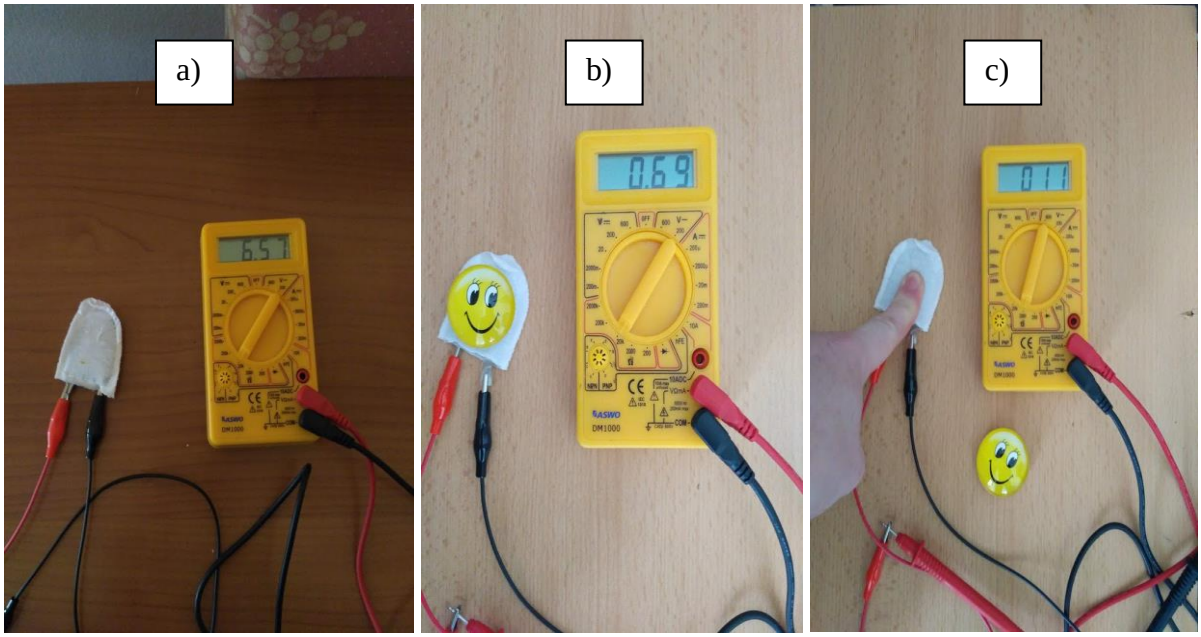


Slika 30. Električni otpori: a) velostata, b) vodljive tkanine

6.1.2. Električna svojstva senzora

Mjerenjem električnog otpora senzora dobiveni su rezultati koji u prosjeku iznose 7 kΩ u neopterećenom stanju. Pritiskom na senzor otpor se znatno smanjuje. Već malim pritiskom bilježe se velike promjene električnog otpora dok se sve jačim pritiskom otpor počinje

mijenjati sve manjom intenzivnošću. Maksimalnim, vlastitim pritiskom prsta izmjerena je vrijednost električnog otpora od 11 Ω (Slika 31.).



Slika 31. Električni otpori senzora: a) neopterećeno stanje, b) predmet postavljen na senzor, c) maksimalan pritisak prstom

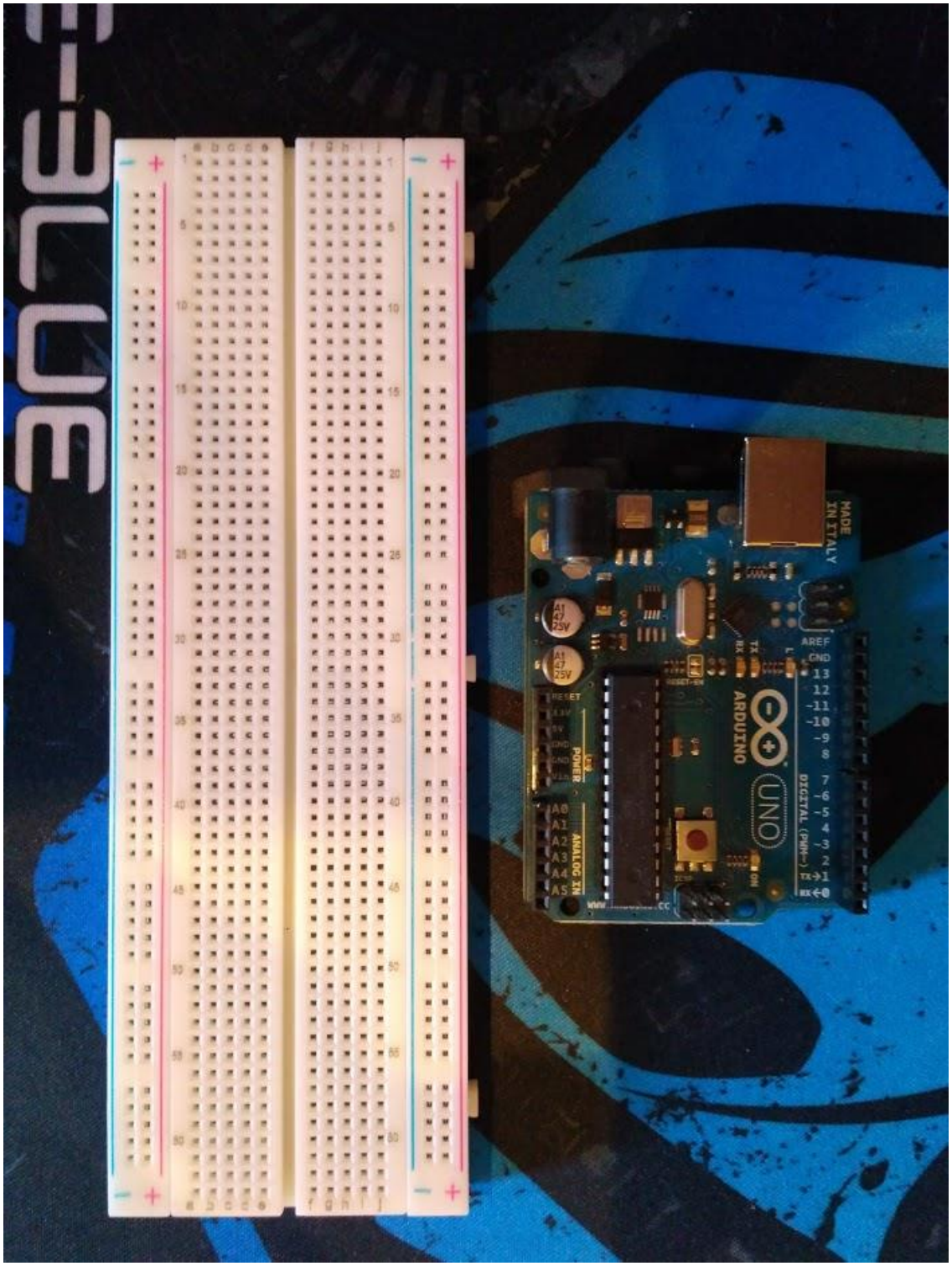
6.1.3. Slika odziva senzora na računalu

Za istovremeni prikaz opterećenja na pojedinim prstima šake potrebno je najmanje pet senzora spojenih u jedan ili više zasebnih strujnih krugova.

U svrhu interpretacije rezultata na računalu, za različite iznose opterećenja, korištena je elektronička pločica Arduino Uno[9] uz popratne hardverske komponente kao što su otpornici, kabeli i dodatna pločica[10] za spajanje komponenti (Slika 32.).

[9] Arduino Uno – elektronička pločica bazirana na mikrokontroleru ATmega328P, opremljena setom digitalnih i analognih ulazno/izlaznih pinova koji se mogu povezivati s različitim ekspanzijskom krugovima. lit?

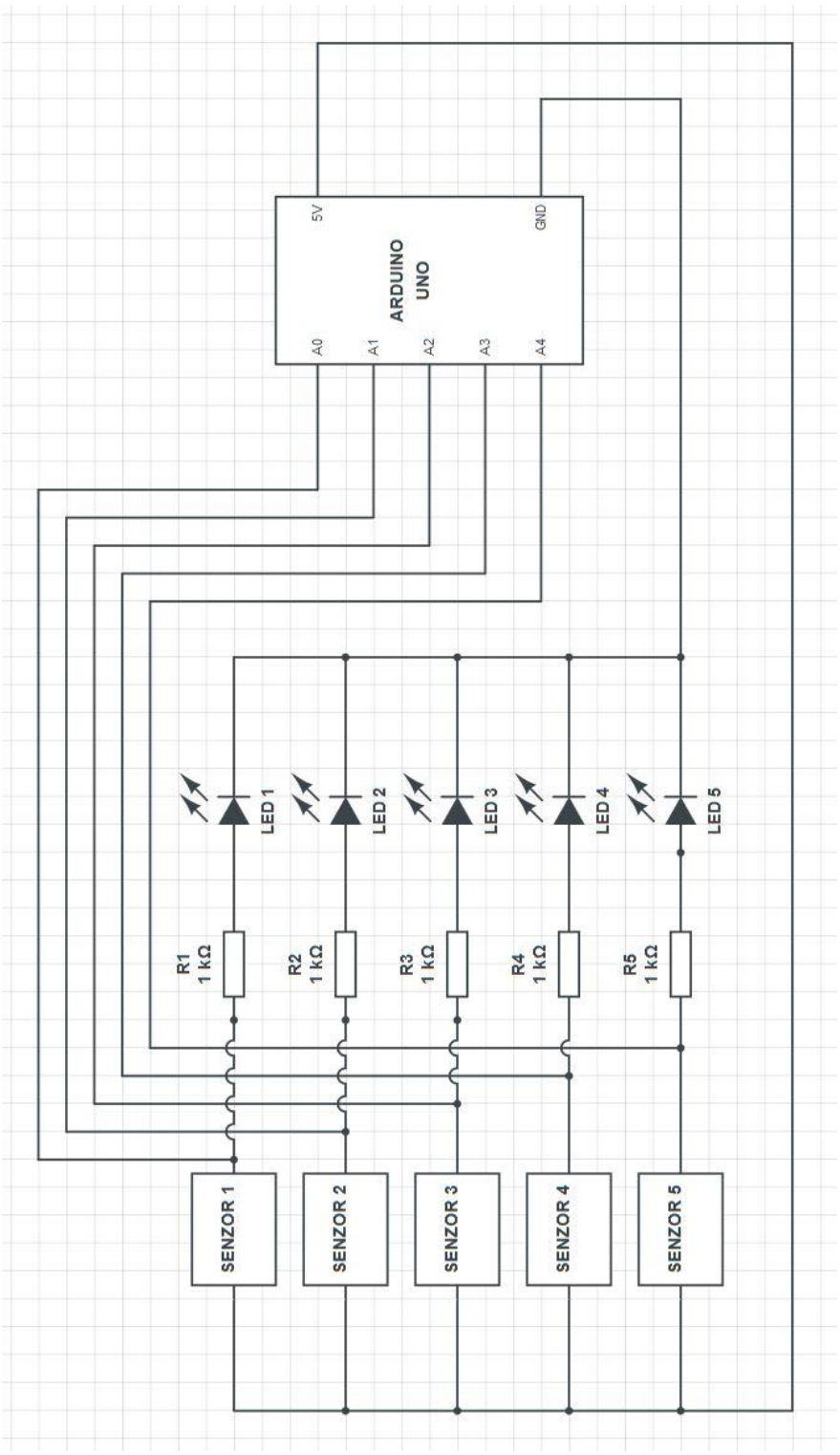
[10] engl. *breadbord*



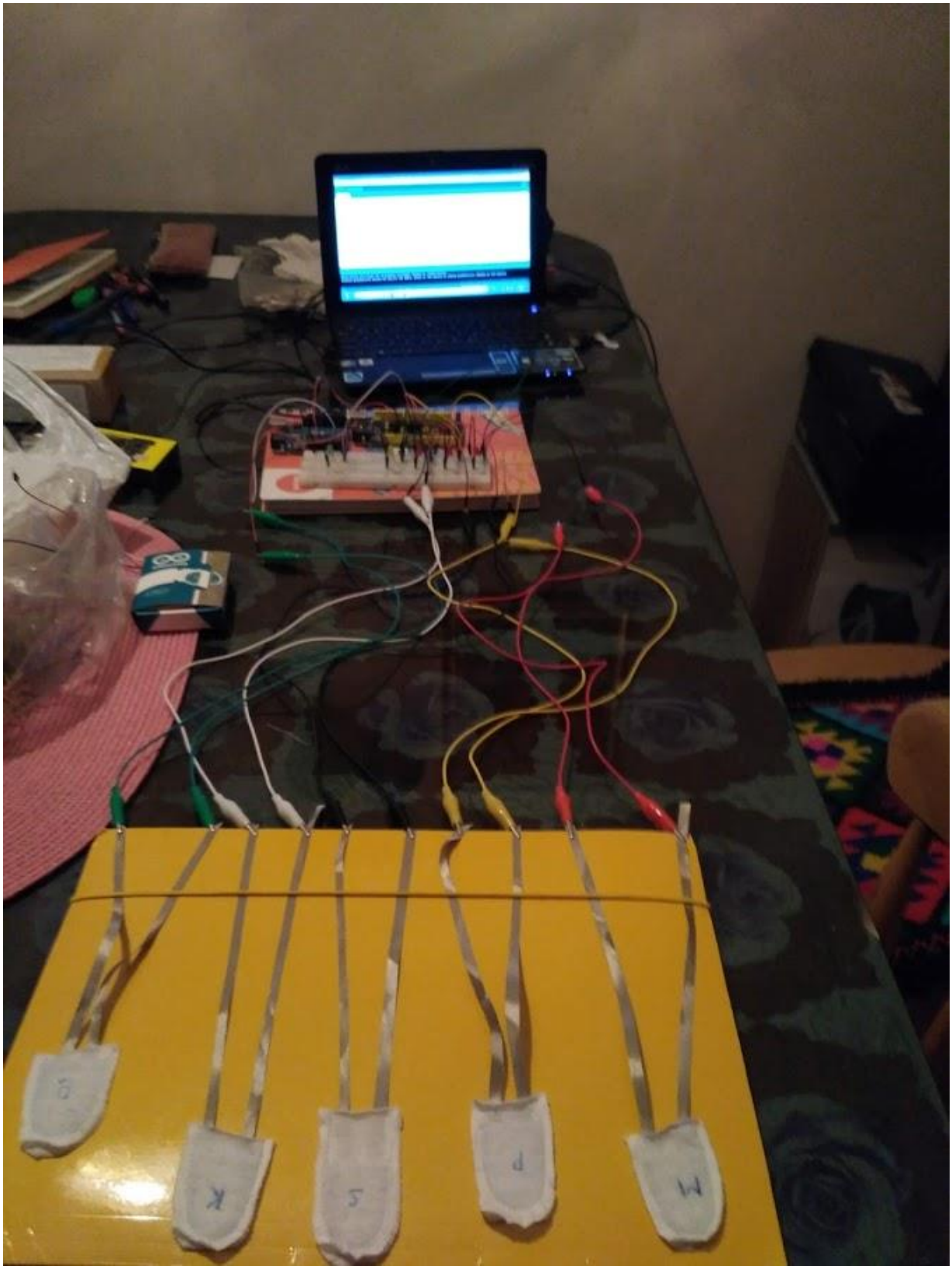
Slika 32. Pločica Arduino uno i *breadboard*

U ovom je slučaju svih pet senzora paralelno spojeno u jedan istosmjerni strujni krug. Izvor napona osiguran je preko pločice Arduino Uno te iznosi 5 V. Pločica je spojena na računalo putem kabela USB kojim se osim komunikacije ostvaruje i napajanje pločice, a samim time i

strujnog kruga. Signal se s pojedinih, paralelno spojenih senzora, dovodi na analogne ulazno/izlazne pinove Arduina te se na računalu interpretira pojedini signal odnosno odziv pojedinog senzora (slike 33. i 34.).



Slika 33. Shematski prikaz strujnog kruga i Arduino Uno



Slika 34. Arduino Uno i spojevi senzora

Za programiranje mikrokontrolera, ostvarivanje komunikacije s računalom i prikaz rezultata na računalu korišten je besplatan softver IDE (*Integrated Development Environment*) koji je lako dostupan na službenoj web stranici proizvođača pločice.[11] Programiranje se ostvaruje u prilagođenoj verziji programskog jezika C++ (Slika 35.).

```
//Definiranje varijabli imena: a, b, c i d tipa integer
int a;
int b;
int c;
int d;
int e;

void setup() {
  //Ovaj dio koda čita se samo jednom
  //Pokretanje serijske komunikacije računala i pločice brzinom od 9600 bita u sekundi
  Serial.begin(9600);
}

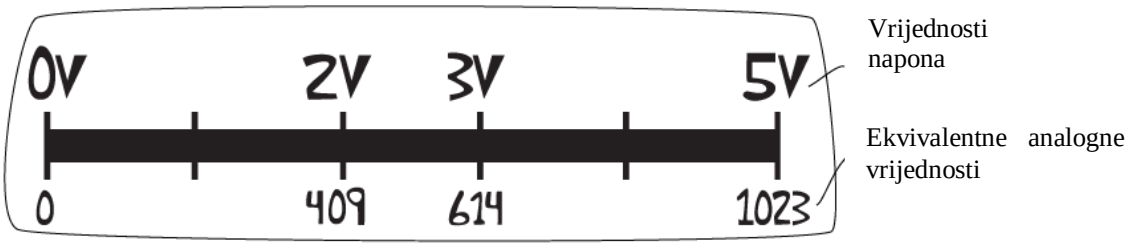
void loop() {
  //Ovaj dio koda se ponavlja
  a=analogRead(A0); //Čitanje vrijednosti napona s analognog pina A0 i dodjeljivanje vrijednosti varijabli a
  delay(50);        //Stanka od 50 ms prije nastavka
  b=analogRead(A1); //Čitanje vrijednosti napona s analognog pina A1 i dodjeljivanje vrijednosti varijabli b
  delay(50);        //Stanka od 50 ms prije nastavka
  c=analogRead(A2); //Čitanje vrijednosti napona s analognog pina A2 i dodjeljivanje vrijednosti varijabli c
  delay(50);        //Stanka od 50 ms prije nastavka
  d=analogRead(A3); //Čitanje vrijednosti napona s analognog pina A3 i dodjeljivanje vrijednosti varijabli d
  delay(50);        //Stanka od 50 ms prije nastavka
  e=analogRead(A4); //Čitanje vrijednosti napona s analognog pina A4 i dodjeljivanje vrijednosti varijabli e


  Serial.print(a);  //Ispis vrijednosti varijable a
  Serial.print(" ");// Ispis praznog mjesta
  Serial.print(b); //Ispis vrijednosti varijable b
  Serial.print(" ");// Ispis praznog mjesta
  Serial.print(c); //Ispis vrijednosti varijable c
  Serial.print(" ");// Ispis praznog mjesta
  Serial.print(d); //Ispis vrijednosti varijable d
  Serial.print(" ");// Ispis praznog mjesta
  Serial.println(e); //Ispis vrijednosti varijable e i prebacivanje u novi red
  delay(50); //Stanka od 50 ms
}
```

Slika 35. Programski kôd za snimanje odziva senzora

[11] https://www.arduino.cc/

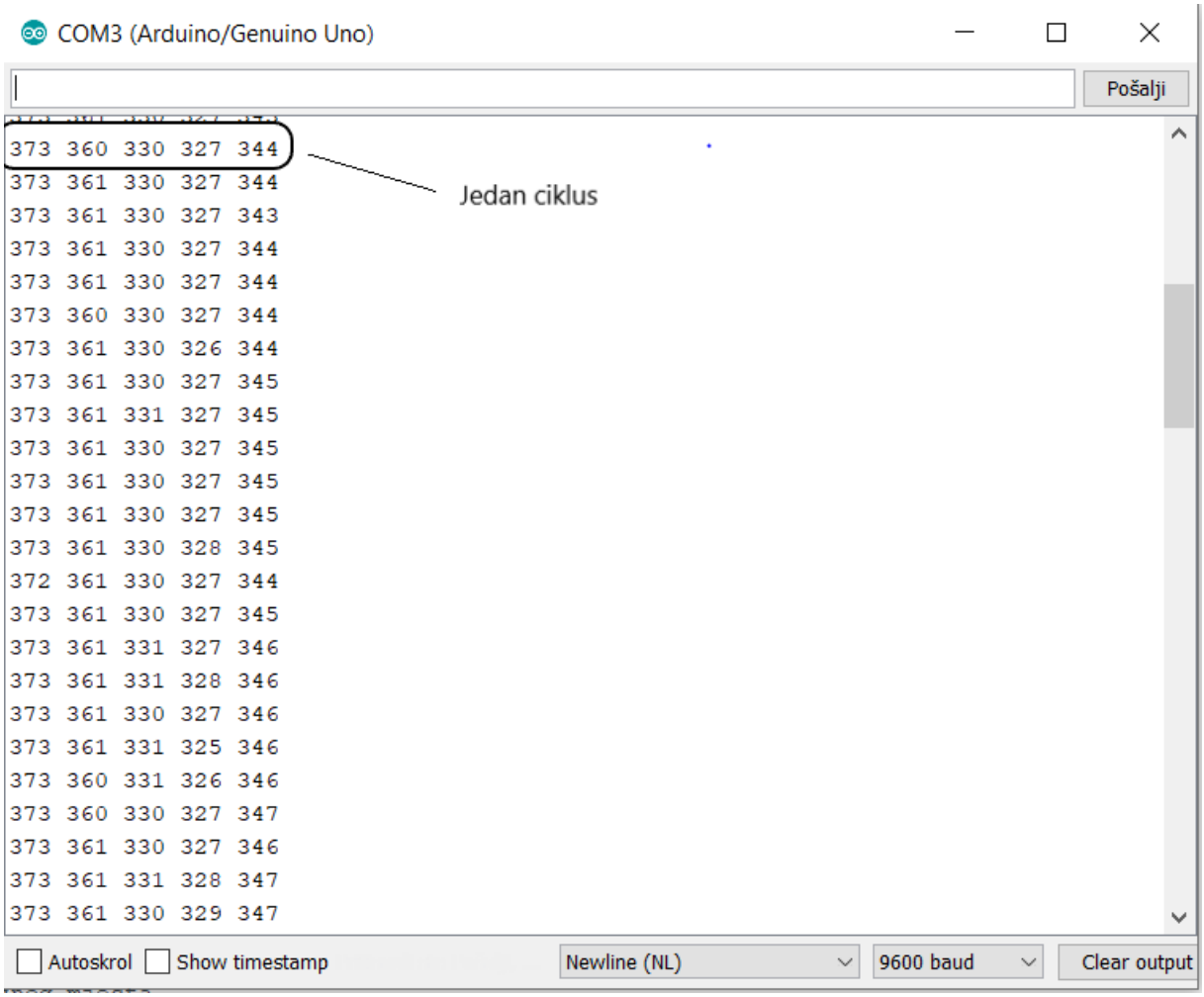
Prikazanim programskim kôdom mikrokontroleru je definirano pet varijabli cjelobrojnog tipa[12]. Potom slijedi naredba za početak serijske komunikacije računala i mikrokontrolera po brzini prijenosa od 9600 bita u sekundi. Očitane vrijednosti napona na analognim ulazima u ovisnosti su o pritisku na senzor, a moguće očitane vrijednosti kreću se od 0 do 5 V. Ove vrijednosti se deset-bitnim analogno-digitalnim pretvaračem (sadržanim u Arduino Unu) pretvaraju u cjelobrojne vrijednosti od 0 do 1023, što je 2^{10} različitih vrijednosti. Kada bi na analognom ulazno/izlaznom pinu očitan napon bio 0 V (Slika 36.), definiranoj varijabli bi se upisao podatak cjelobrojnog tipa vrijednosti nula, a kada bi očitana vrijednost bio napon od 5 V varijabli bi se dodijelio cjelobrojni podatak cjelobrojnog tipa, vrijednosti 1023. Dakle, svaku vrijednost napona između 0 i 5 V analogno-digitalni pretvarač će pretvoriti u podatak cjelobrojnog tipa od 0 do 1023, naprimjer 2,5 V predstavljalo bi podatak cjelobrojnog tipa, vrijednosti 511.



Slika 36. Odnos vrijednosti napona i analognih vrijednosti [22]

Ove vrijednosti u rasponu od 0 do 1023 dodjeljuju se prije definiranim varijablama (Slika 35.) te se vrijednosti svakog od pet senzora u jednom ciklusu zapisuju u jednoj liniji teksta (Slika 37.). Dakle, vrijednosti napona za sljedeći ciklus na isti se način zapisuju u novom retku.

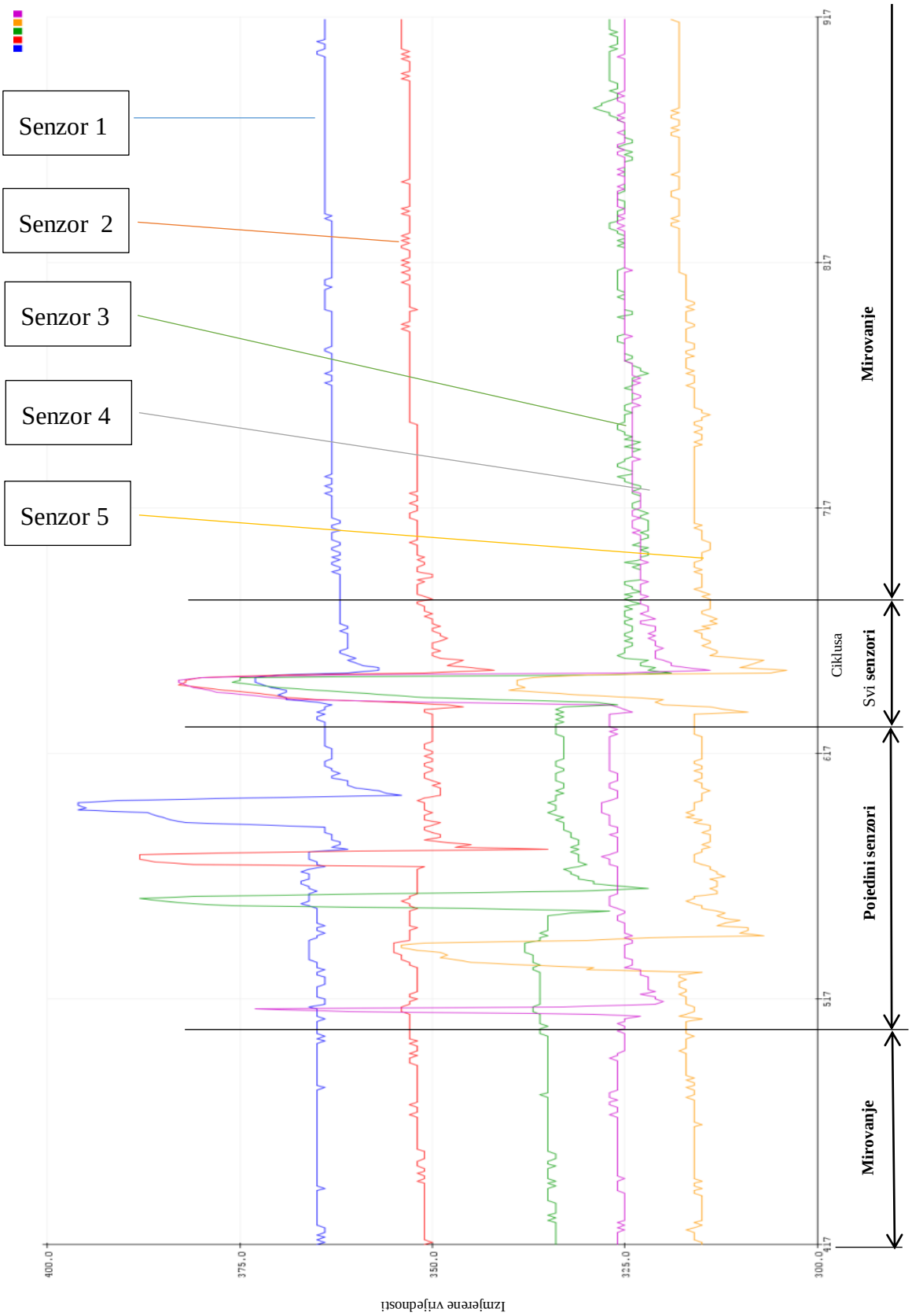
[12] eng. *integer*, raspon vrijednosti od –32768 do 32768.



Slika 37. Prikaz zapisa vrijednosti varijabli za svaki senzor u više ciklusa

Za zorniji prikaz ponašanja pojedinog senzora dan je prikaz u obliku grafa (Slika 38.). Horizontalna os predstavlja broj ciklusa ili vrijeme, a vertikalna os vrijednost varijabli. Iscrtavanjem vrijednosti varijabli za svaki se ciklus na opisanom grafu stječe vizualni uvid o ponašanju pojedinih senzora.

Slika 38. daje grafički prikaz ponašanja senzora u vremenu.



Slika 38. Grafički prikaz odziva senzora

U stanju mirovanja svih pet senzora, zanemarujući sitno poskakivanje u plus i minus, daje konstantnu vrijednost napona. Potom slijedi pojedinačni podražaj na svaki senzor u kratkom vremenskom intervalu ili manjem broju ciklusa te na kraju podražaj svih senzora od jednom što je vidljivo odgovarajućim porastima vrijednosti varijabli na grafu.

6.2. Sastavljanje podatkovne rukavice

Prethodno opisane taktilne senzore potrebno je smjestiti na rukavicu na način da se na jagodičnom dijelu pojedinog prsta nalazi po jedan senzor. Kako bi senzori radili i bili od koristi potrebno ih je povezati u strujni krug spojen na elektroničku pločicu Arduino Uno te time ostvariti i povezanost računala i senzora, odnosno računala i podatkovne rukavice.

U svrhu fizičke realizacije podatkovne rukavice korištena je vrtna rukavica, lako dostupna u mnogim trgovinama, na kojoj su vrućim ljepilom pričvršćeni senzori. Senzori su putem elektrovodiča načinjenih od izrezane vodljive tkanine spojeni na metalne kopče („drukere“) kojim je ostvareno jednostavno spajanje rukavice na kabel koji povezuje rukavicu i pločicu Arduino Uno (Slika 39.). Vodljivi dijelovi rukavice izolirani su ljepljivom, crvenom trakom te na pojedinim mjestima vrućim ljepilom (Slika 40.).



Slika 39. Gornja strana rukavice prije i poslije stavljanja senzora



Slika 40. Donja strana ruakvice prije i poslije stavljanja senzora

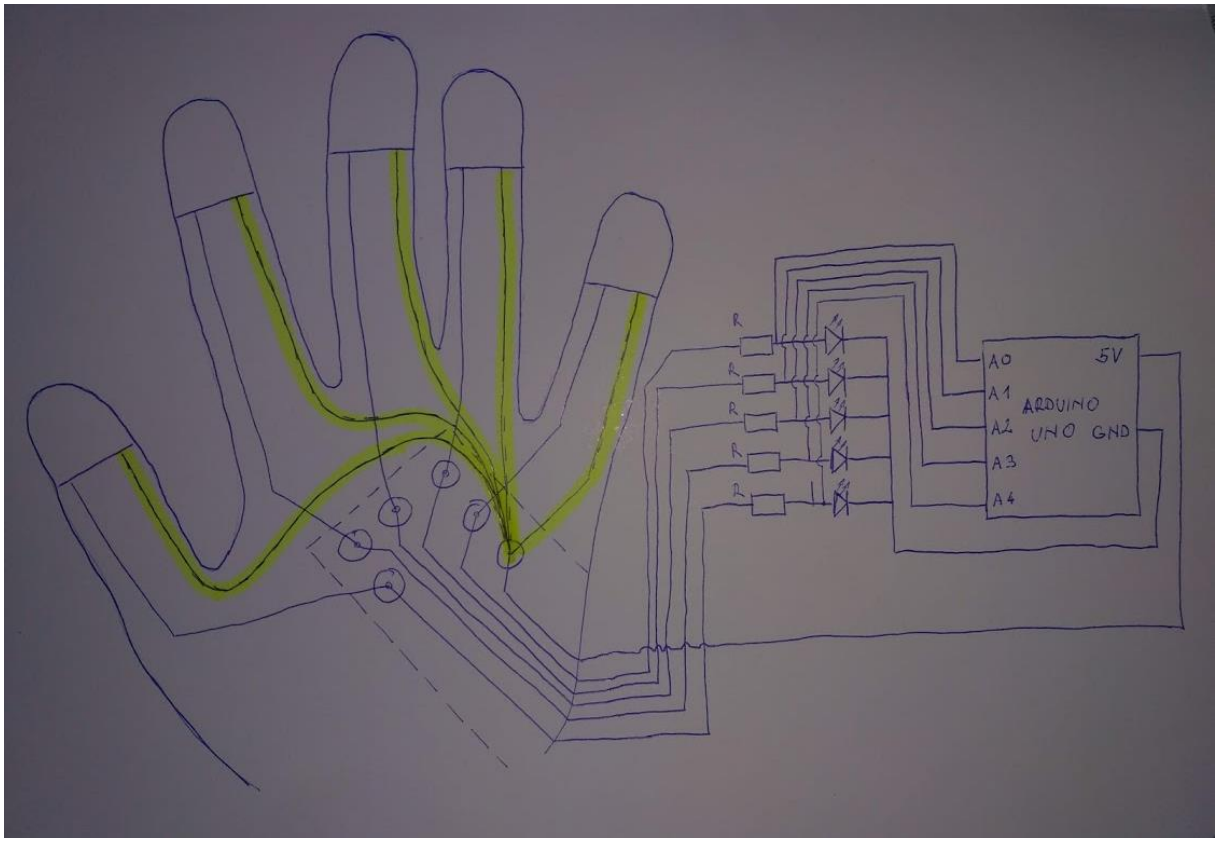
Za potrebe spajanja same rukavice, odnosno njenih senzora, u strujni krug izrađen je kabel s više međusobno odvojenih i izoliranih žica (Slika 41.). Kabel se na rukavicu spaja putem metalnih kopči, a na pločicu Arduino Uno umetanjem „golog“ kraja žice u *breadboard*. Kabelom je ostvarena fleksibilnost te mogućnost djelovanja podatkovne rukavice na većem radnom prostoru.



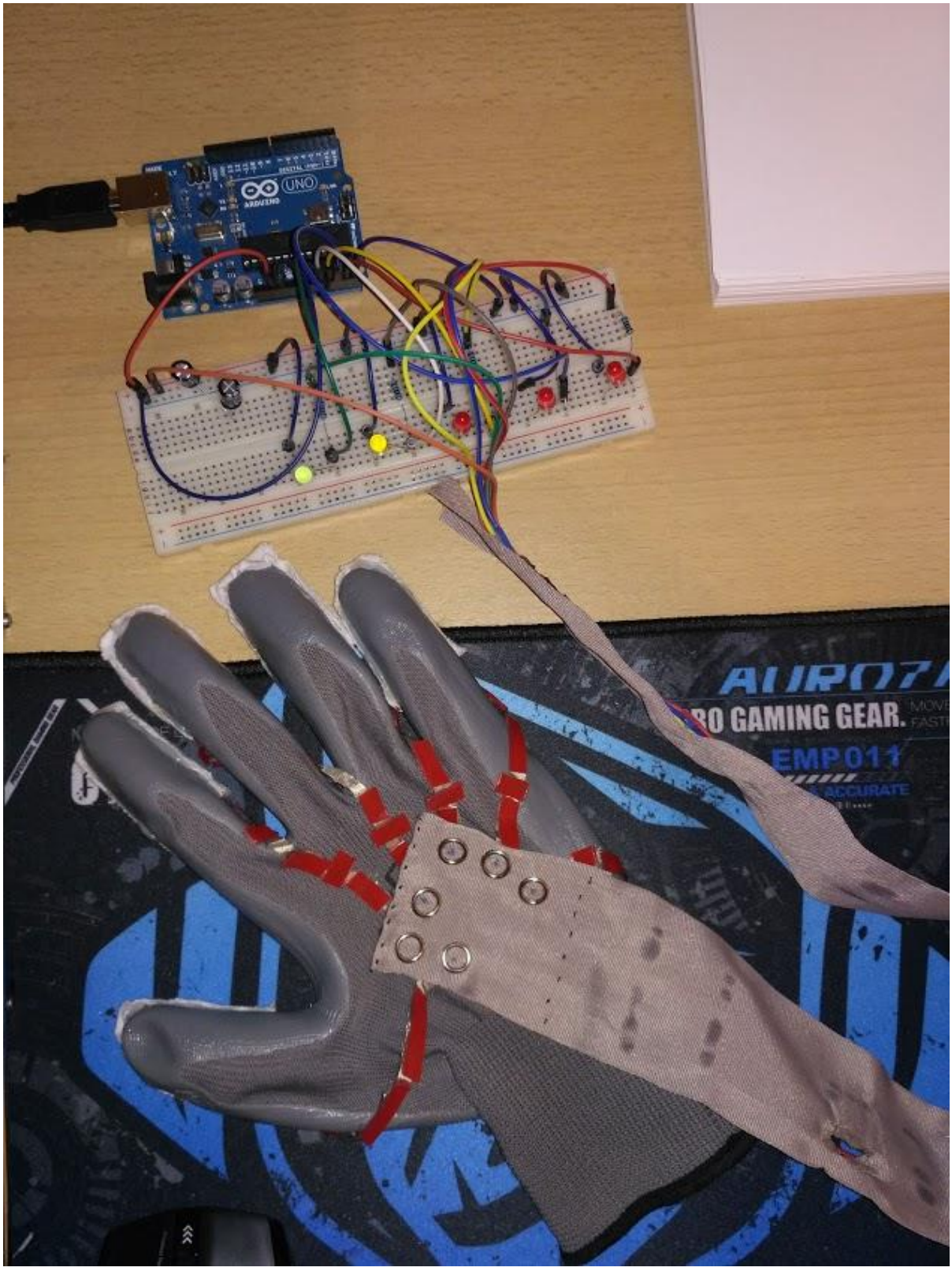
Slika 41. Kabel za rukavicu

Senzori smješteni na rukavicu spojeni su u jedan strujni krug i pločicu Arduino Uno kao i kod prethodnog spajanja samih senzora (Slika 33.). S obzirom da su senzori ovaj puta smješteni na rukavicu, fizički, strujni krug je izveden malo drugačije. Jedna metalna kopča spojena je putem pločice Arduino Uno na izvor napona od 5 V te je preko iste kopče spojena jedna vodljiva nožica svakog senzora. Druga nožica svakog pojedinog senzora dalje se paralelno spajaju u strujni krug. Svakom senzoru je u seriju dodan otpornik (R) od 1 kΩ te po jedna LED[13] koja svijetli intenzivnošću proporcionalnom pritisku prsta. Iz svakog paralelnog spoja dovodi se signal na analogne ulazno/izlazne pinove (A0, A1, A2, A3, A4). Cijeli strujni krug zatvara se spajanjem u masu (GND) (slike 42. i 43.).

[13] LED – eng. *Light Emitting Diode*



Slika 42. Shematski prikaz spoja rukavice u strujni krug



Slika 43. Podatkovna rukavica spojena na pločicu Arduino Uno izrađenim kabelom

6.3. Interpretacija rada rukavicom na računalu

Za interpretaciju djelovanja senzora na računalu korišteni su softveri MATLAB[14] i prije navedeni IDE (Slika 35.). Programom IDE ostvarena je komunikacija računala i pločice Arduino Uno te ujedno i poslan program mikrokontroleru za snimanje informacija koje dolaze na definirane analogne ulazno/izlazne pinove. Snimljeni podatci su potom obrađivani i interpretirani programom MATLAB, a isti program je korišten također i za upis vrijednosti u u pločicu Arduino Uno.

6.3.1. Uspostavljanje komunikacije

Za programiranje mikrokontolera za izvršavanje željene funkcije najprije su definirane programom IDE varijable koje će biti korištene. Varijable su cjelobrojnog tipa te su im dodijeljena imena. Potom se pokreće serijska komunikacija računala i pločice Arduino Uno preko definiranog sučelja USB brzinom prijenosa od 9600 bita u sekundi.

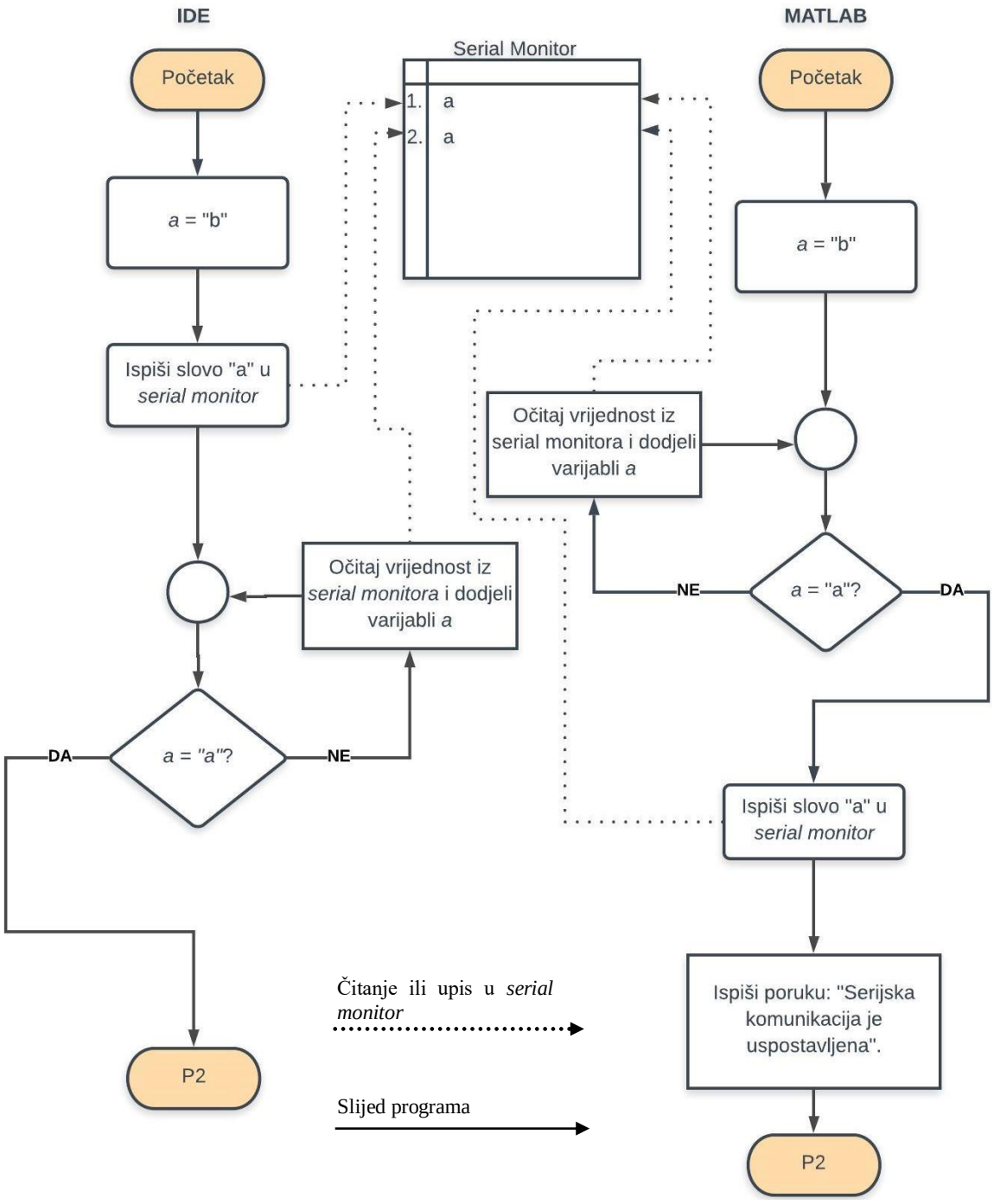
Slijedi provjera komunikacije s MATLAB-om na način da se putem softvera IDE pošalje naredba pločici Arduino Uno da u *serial monitor*[15] upiše slovo a . Potom se u IDE prethodno definiranoj varijabli imena *a* koja je znakovnog tipa[16] dodjeljuje vrijednost b. Nakon toga slijedi petlja kojoj je postavljen uvjet da se ponavlja sve dok varijabla *a* ne poprimi vrijednost znakovnog tipa a. Unutar petlje ponavljajući se čita upisana vrijednost u sljedećem retku serial monitora i dodjeljuje se varijabli *a* (Slika 44.). Potom se ručno u MATLAB-u pokreće program kojim se poziva funkcija za uspostavljanje međusobne komunikacije. Ovom funkcijom MATLAB-u je definiran port USB na koji je spojen Arduino Uno i definiran je početak komunikacije kao i parametri komunikacije. Funkcijom se u MATLAB-u varijabli pod imenom *a* dodjeljuje vrijednost znakovnog tipa b. Potom slijedi petlja koja se ponavlja

[14] MATLAB – programski jezik i interaktivna okolina za numeričko i matrično računanje, vizualizaciju i programiranje

[15] *Serial monitor* (Slika 37.) prozor na računalu u koji ispisuje vrijednosti koje pločica Arduino Uno šalje računalu putem serijske komunikacije

[16] eng. *character*

sve dok je vrijednost varijable *a* različita od slova a. Unutar petlje čita se vrijednost u prvom retku *serial monitora* te ako se pročita slovo a izlazi se iz petlje i ispisuje se u skočnom prozoru: „Serijska komunikacija je uspostavljena“, a u drugom retku *serial monitora* ujedno se i upisuje vrijednost a što je uvjet za izlazak iz petlje programa pisanog u IDE (Slika 44.). Programski kod za uspostavljanje komunikacije prikazan je na slikama 45. do 47.



Slika 44. Dijagram toka za uspostavljanje komunikacije

```
//Definicija varijabli:
int mod = -1;
int vrijednost1;
int vrijednost2;
int vrijednost3;
int vrijednost4;
int vrijednost5;


void setup() {

//Započinjanje serijske komunikacije s računalom po 9600bps:
  Serial.begin(9600);

//Provjera serijske veze s računalom
  Serial.println("a"); //Ispiši u serial monitoru u jednom retku
                        //samo jedan element, slovo a
  char a = 'b'; //Definiranje varijeble a formata character i dodjeljivanje
                //vrijednosti različitog od slova a
    while (a != 'a') //Dok je god varijabla a različita od slova a čitaj
                     //vrijednost zapisanu u serial monitoru,
                     //kada poprimi slovo a, izađi iz petlje
      {
        a = Serial.read();  //varijabla a poprima vrijednost iz text datoteke
      }
}
```

Slika 45. Programski kôd u IDE za definiranje komunikacije

```
1 -    clc; %Obriši sve s comand window
2 -    close all; %Zatvori sve otvorene prozore
3 -    clear all; %Obriši sve varijable i ostalo
4
5      %Definiranje porta
6 -    comPort='COM3';
7
8      %Testiranje serijske komunikacije
9 -    if (~exist('serialFlag','var'))
10 -       [senzor.s, serialFlag] = Uspostavljanje_veze(comPort);
11 -   end
```

Slika 46. Programski kôd za pozivanje funkcije za uspostavljanje komunikacije u MATLAB-u

```
function [s, flag] = Uspostavljanje_veze(comPort)

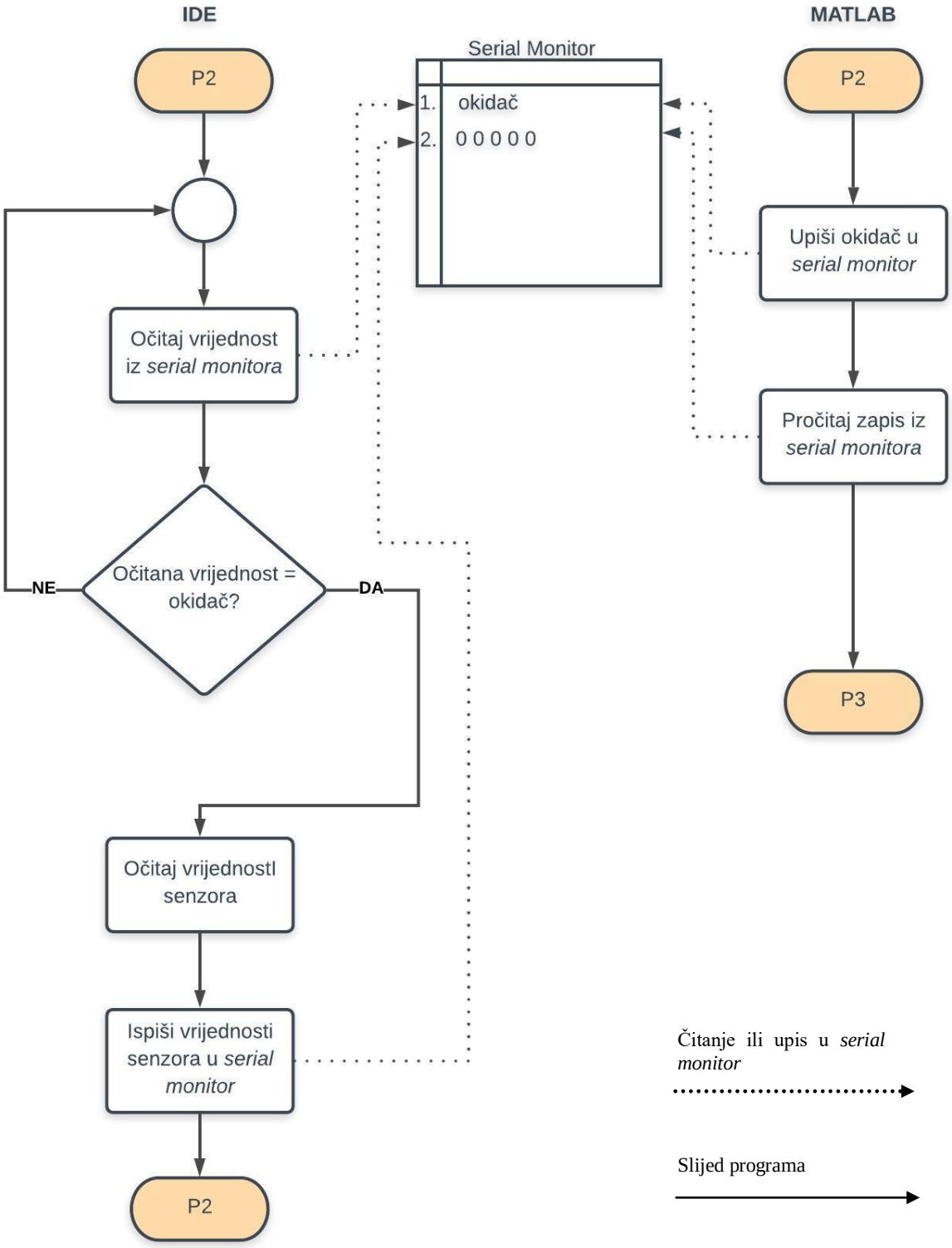
flag = 1;
s = serial(comPort);
set(s, 'DataBits', 8);
set(s, 'StopBits', 1);
set(s, 'BaudRate', 9600);
set(s, 'Parity', 'none');

fopen(s); %Otvaranje porta
a='b'; %Varijabli a se dodjeljuje vrijednost formata %c
while (a ~= 'a') %Dok varijabla ne poprimi vrijednost "b" ulazi se u petlju
    a=fread(s, 1,'uchar');  %Čita se vrijednost varijable a iz serial monitora,
                            %vrijednost je veličine 1 elementa i uchar preciznosti
                            %i dodjeljuje se varijabli a
end
if (a == 'a') %Ako varijabla a poprimila vrijednost 'a'
    disp('Serijska komunikacija'); %Ispiši poruku: Serijska komunikacija
end
fprintf(s, '%c', 'a'); %U serijski port upisuje se slovo "a" varijable a, formata c.
mbox=msgbox('Serijska komunikacija je uspostavljena.'); uiwait(mbox); %Ispiši poruku
          %da je uspostavljena serijska komunikacija i čekaj dok se ne zatvori prozor
fscanf(s, '%u'); %Pročitaj zapisanu vrijednost u serijskom portu i pretvori ju u decimalni zapis
end
```

Slika 47. Programski kôd funkcije za uspostavljanje komunikacije u MATLAB-u

6.3.2. Čitanje i ispisivanje vrijednosti senzora

Po završetku uspostavljanja komunikacije MATLAB-a i pločice Arduino Uno mikrokontroler se postavlja u stanje u kojem čeka okidač iz MATLAB-a za početak čitanja i ispisa vrijednosti senzora (Slika 48.). Kad se iz MATLAB-a pošalje odgovarajući okidač, mikrokontroler u jednom ciklusu iščita vrijednosti svih senzora i ispisuje ih u jednom retku *serial monitora* (Slika 37.). Ispisane vrijednosti senzora MATLAB koristi za interpretaciju na računalu. Spomenuti okidač za snimanje stanja senzora koji šalje MATLAB poziva se funkcijom koja slijedi u programskom kôdu unutar MATLAB-a nakon izvršavanja uspostavljanja komunikacije. Programski kôd unutar IDE i MATLAB-a vidljiv je na slikama 49. do 51.



Slika 48. Dijagram toka za očitavanje i ispis vrijednosti senzora

```
void loop() {

  if (Serial.available()>0) //Ako postoje vrijednosti za čitanje s serijskog porta
  {
    mod = Serial.read(); //Varijabla mod poprima očitanu vrijednost
    switch (mod) //Vrti petlju dok se ne ipune dolje navedeni uvjeti
    {
      case 'F': //Kada varijabla mod poprimi vrijednost F
      vrijednost1 = analogRead(A0); //Varijabla vrijednost1 poprima očitanu vrijednost s pina A0
      delay(50);                    //Čekanje 50 ms
      vrijednost2 = analogRead(A1); //Varijabla vrijednost2 poprima očitanu vrijednost s pina A1
      delay(50);                    //Čekanje 50 ms
      vrijednost3 = analogRead(A2); //Varijabla vrijednost3 poprima očitanu vrijednost s pina A2
      delay(50);                    //Čekanje 50 ms
      vrijednost4 = analogRead(A3); //Varijabla vrijednost4 poprima očitanu vrijednost s pina A3
      delay(50);                    //Čekanje 50 ms
      vrijednost5 = analogRead(A4); //Varijabla vrijednost5 poprima očitanu vrijednost s pina A4
      delay(50);                    //Čekanje 50 ms
      Serial.print(vrijednost1); //Ispiši vrijednost varijable vrijednost1 u serial monitor
      Serial.print(" ");         //Ispiši razmak
      Serial.print(vrijednost2); //Ispiši vrijednost varijable vrijednost2 u serial monitor
      Serial.print(" ");         //Ispiši razmak
      Serial.print(vrijednost3); //Ispiši vrijednost varijable vrijednost3 u serial monitor
      Serial.print(" ");         //Ispiši razmak
      Serial.print(vrijednost4); //Ispiši vrijednost varijable vrijednost4 u serial monitor
      Serial.print(" ");         //Ispiši razmak
      Serial.println(vrijednost5); //Ispiši vrijednost varijable vrijednost5 u serial monitor
                                   //i prebaci se u sljedeći red
      delay(50); //Čekaj 50 ms
      break;
    }
  }
}
```

Slika 49. Programski kôd u programu IDE za očitavanje vrijednosti senzora po okidaču iz MATLAB-a

```
162 -    [masa]=Ocitaj_vrijednost(senzor);
163
```

Slika 50. Programski kôd u MATLAB-u za pozivanje funkcije za očitavanje stanja senzora

```
1      function [vrijednost] =Ocitaj_vrijednost(analogni_pin)
2 -      fprintf(analogni_pin.s, 'F'); %Upiši u text datoteku slovo F
3 -      vrijednost=fscanf(analogni_pin.s, '%d'); %Očitaj vrijednost iz serial monitora
4 -    end
```

Slika 51. Programski kôd funkcije u MATLAB-u za slanje okidača i očitavanje stanja senzora

Nakon očitanja vrijednosti senzora u jednom ciklusu, snimljeni podatak se programom MATLAB ispisuje i iscrtava u zasebnom prozoru.

6.3.3. Kalibracija senzora i računanje vrijednosti opterećenja

S obzirom da su senzorom snimljene vrijednosti zapravo neki cjelobrojni podatci u rasponu od 0 do 1023 potrebno im je dati smisao odnosno pretvoriti ih u iznos odgovarajuće sile. U tu svrhu provodi se kalibracija odnosno senzori se opterete poznatim iznosima opterećenja te se snimljenim, cjelobrojnim vrijednostima pridružuju stvarne vrijednosti opterećenja. Kalibracija će biti bolja i kasnije očitane vrijednosti točnije ukoliko se uzme što veći broj različitih, poznatih opterećenja kojima se definiraju snimljene vrijednosti. Za sva ostala opterećenja kojima nije definirana vrijednost, računalo će na temelju provedene aproksimacije iz izmjerenih vrijednosti za poznata opterećenja dati pretpostavku o iznosu.

Osjetljivost senzora odnosno promjena otpora u ovisnosti o promjeni opterećenja, međutim, nije konstantna jer je i statička karakteristika otpora nelinearna ($R=(k/p)+R_u$, gdje je k/p kontaktni otpor, a R_u unutarnji otpor). Električna otpornost senzora smanjuje se s porastom normalne sile koja djeluje na senzor prema eksponencijalnoj jednadžbi [13]:

$$R_m = Ke^{-F/\beta}. \tag{1}$$

S obzirom na prethodno navedenu relaciju (1) osjetljivost otpora na mjerenu silu je:

$$\frac{dR_i}{dF} = -\frac{K}{\beta}e^{-F/\beta} = -\frac{R_m}{\beta}. \tag{2}$$

Veličine K i β imaju konstantne vrijednosti, a budući da su u okviru ovog rada vrijednosti konstanti nepoznate provodi se aproksimacija prema snimljenim vrijednostima za pojedina poznata opterećenja. U tu svrhu koristi se MATLAB-ova funkcija *polyfit* za računanje koeficijenata Lagrangeovog interpolacijskog polinoma s odabranim stupnjem polinoma 3 (Slika 52.). Vrijednosti opterećenja za pojedine izmjere senzora potom se izračunavaju MATLAB-ovom funkcijom *polyval* (Slika 53.).

```
85 -    P1=polyfit(Palac, mase, 3);
86 -    P2=polyfit(Kaziprst, mase, 3);
87 -    P3=polyfit(Srednji, mase, 3);
88 -    P4=polyfit(Prstenjak, mase, 3);
89 -    P5=polyfit(Mali, mase, 3);
```

Slika 52 Programski kôd pozivanja MATLAB-ove funkcije *polyfit*

```
164 -   masa1=polyval(P1, masa(5,1));
165 -   masa2=polyval(P2, masa(4,1));
166 -   masa3=polyval(P3, masa(3,1));
167 -   masa4=polyval(P4, masa(2,1));
168 -   masa5=polyval(P5, masa(1,1));
```

Slika 53 Programski kôd pozivanja MATLAB-ove funkcije *polyval*

Uzorkovanje se provodi na temelju poznatih iznosa masa kojim se opterećuje senzor te se za poznate mase navedenim funkcijama u MATLAB-u provodi aproksimacija. Uzorkovanje se u MATLAB-u provodi unutar *for* petlje gdje se svaki puta poziva funkcija za očitavanje trenutne vrijednosti na senzoru (Slika 54., Slika 55.). Funkcije za očitavanje vrijednosti kod kalibracije jednake su onima opisanim u poglavlju o čitanju i ispisu vrijednosti (6.3.2.).

```
29      %Kalibracija senzora
30 -    mase=[0 250 300 350 400 450 500 550 600 700 750 800 850 900 1000]; %grama
31 -    m1=zeros(length(mase), 1)';
32 -    m2=zeros(length(mase), 1)';
33 -    m3=zeros(length(mase), 1)';
34 -    m4=zeros(length(mase), 1)';
35 -    m5=zeros(length(mase), 1)';
36
37
38
39      %Pročitaj vrijednost za svaku masu i dodjeli joj odgovarajucu silu
40 -    for i=2:length(mase)
41 -        mbox=msgbox(['Optereti palac masom od' num2str(mase(i)) 'g']); uiwait(mbox);
42 -        m1(i) = Ocitaj_vrijednost_palac(senzor);
43 -       while (m1(i) < m1(i-1)) || m1(i) == 0;
44 -           m1(i) = Ocitaj_vrijednost_palac(senzor);
45 -       end
46 -    end
47 -    Palac=m1;
48
49 -    for i=2:length(mase)
50 -        mbox=msgbox(['Optereti kažiprst masom od' num2str(mase(i)) 'g']); uiwait(mbox);
51 -        m2(i) = Ocitaj_vrijednost_kaziprst(senzor);
52 -       while (m2(i) < m2(i-1)) || m2(i) == 0;
53 -           m2(i) = Ocitaj_vrijednost_kaziprst(senzor);
54 -       end
55 -    end
56 -    Kaziprst=m2;
```

Slika 54 Programski kôd u MATLAB-u za provođenje kalibracije 1

```
for i=2:length(mase)
    mbox=msgbox(['Optereti srednji prst masom od' num2str(mase(i)) 'g']); uiwait(mbox);
    m3(i) = Ocitaj_vrijednost_srednji(senzor);
   while (m3(i) < m3(i-1)) || m3(i) == 0;
      m3(i) = Ocitaj_vrijednost_srednji(senzor);
   end
end
Srednji=m3;

for i=2:length(mase)
    mbox=msgbox(['Optereti prstenjak masom od' num2str(mase(i)) 'g']); uiwait(mbox);
    m4(i) = Ocitaj_vrijednost_prstenjak(senzor);
   while (m4(i) < m4(i-1)) || m4(i) == 0;
      m4(i) = Ocitaj_vrijednost_prstenjak(senzor);
   end
end
Prstenjak=m4;

for i=2:length(mase)
    mbox=msgbox(['Optereti mali prst masom od' num2str(mase(i)) 'g']); uiwait(mbox);
    m5(i) = Ocitaj_vrijednost_mali(senzor);
   while (m5(i) < m5(i-1)) || m5(i) == 0;
      m5(i) = Ocitaj_vrijednost_mali(senzor);
   end
end
Mali=m5;
```

Slika 55 Programski kôd u MATLAB-u za provođenje kalibracije 2

Vrijednosti kalibracije za senzor na kažiprstu

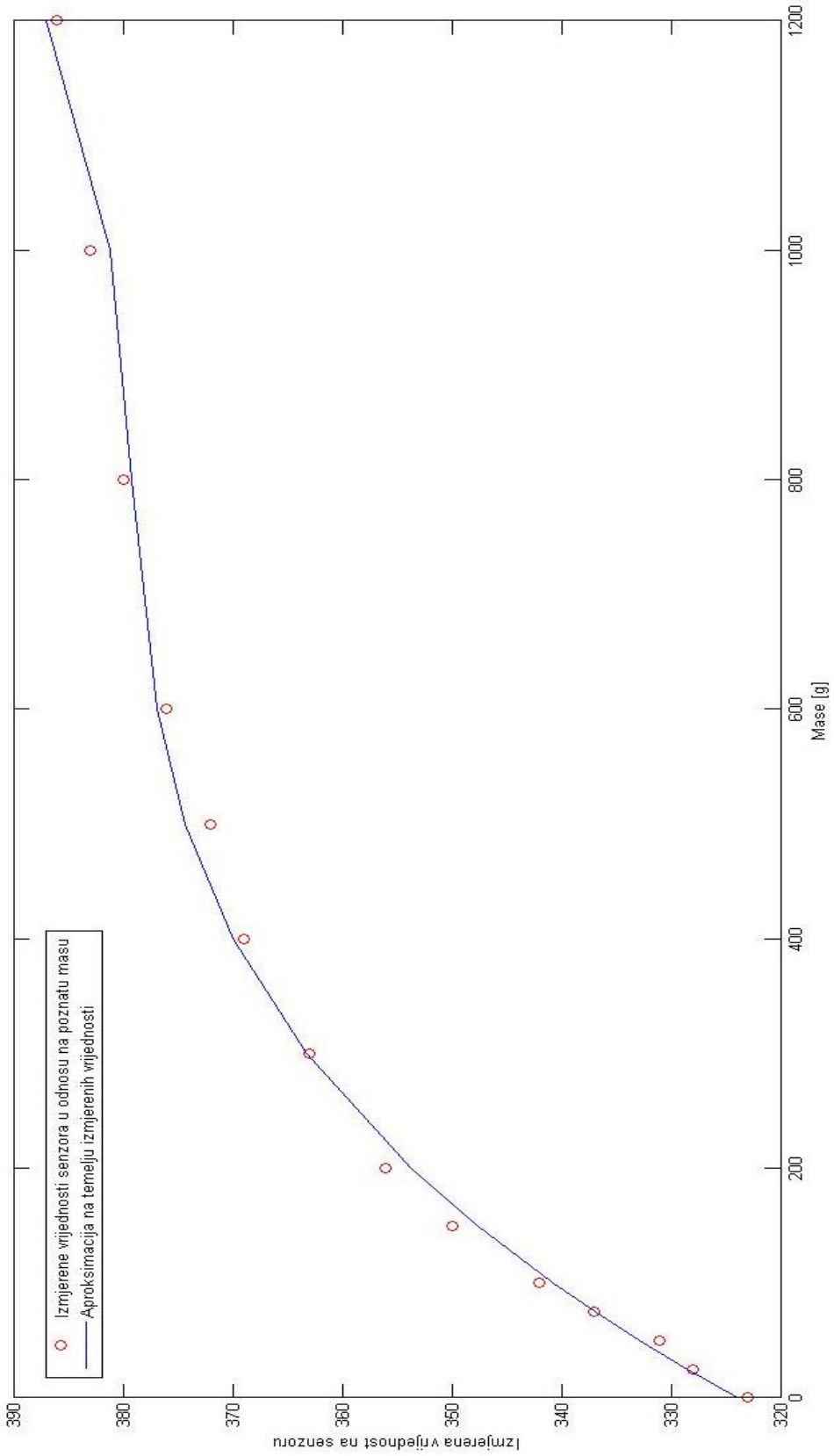
Za interpretaciju aproksimirane funkcije koja predstavlja odnos vrijednosti na senzoru i vrijednosti opterećenja zabilježen je primjer za kažiprst. Pet su puta izmjerene vrijednosti na senzoru za pojedina poznata opterećenja. Iz izmjerenih uzoraka izračunate su prosječne vrijednosti na temelju kojih će se provesti aproksimacija funkcije koja opisuje odnos vrijednosti na senzoru i mase kojom je opterećen senzor. Tablica 3. prikazuje vrijednosti senzora na kažiprstu za pet ponovljenih mjerenja za poznate iznose opterećenja. Prikaz izmjerenih vrijednosti i aproksimirane funkcija dan je na slikom 56.

Tablica 3. Usporedba rezultata ponavljanih mjerenja kažiprsta

Poznate mase opterećenja senzora [g]														
	0	25	50	75	100	150	200	300	400	500	600	800	1k	1.2k
N	IZMJERENE VRIJEDNOSTI NA SENZORIMA													
1.	320	328	333	338	345	349	354	357	360	363	365	368	369	372
2.	320	332	336	343	345	351	356	362	366	369	373	377	379	380
3.	323	328	331	337	342	350	356	363	369	372	376	380	383	386
4.	322	327	333	339	344	350	356	361	363	366	369	374	377	377
5.	320	336	339	346	348	354	355	359	362	366	366	368	369	371
P	321	330	334	341	345	351	355	360	364	367	370	373	375	377

N – broj mjerenja

P – prosječna vrijednost



Slika 56. Aproksimacija na temelju izmjerenih vrijednosti i poznatih masa na kažiprst

7. EKSPERIMENTI PODATKOVNOM RUKAVICOM

Podatkovnom rukavicom razvijenom u okviru ovog rada izveden je niz eksperimenata. Snimljeni su rezultati, odnosno aktivna opterećenja na pojedinim prstima kod rukovanja određenim objektima ili alatima. Eksperimenti su provedeni prije svega radi dobivanja okvirnih informacija o iznosima opterećenja u svrhu stjecanja uvida o mogućnostima poboljšavanja nekog načina rada ili izvedbe alata. Nadalje, snimljenim rezultatima pokušat će se uvidjeti sličnosti opterećenja na pojedinim prstima pri rukovanju objektima ili alatima sličnih oblika, ali različitih dimenzija ili masa.

7.1. Snimanje pokreta opisanih MTM sustavom

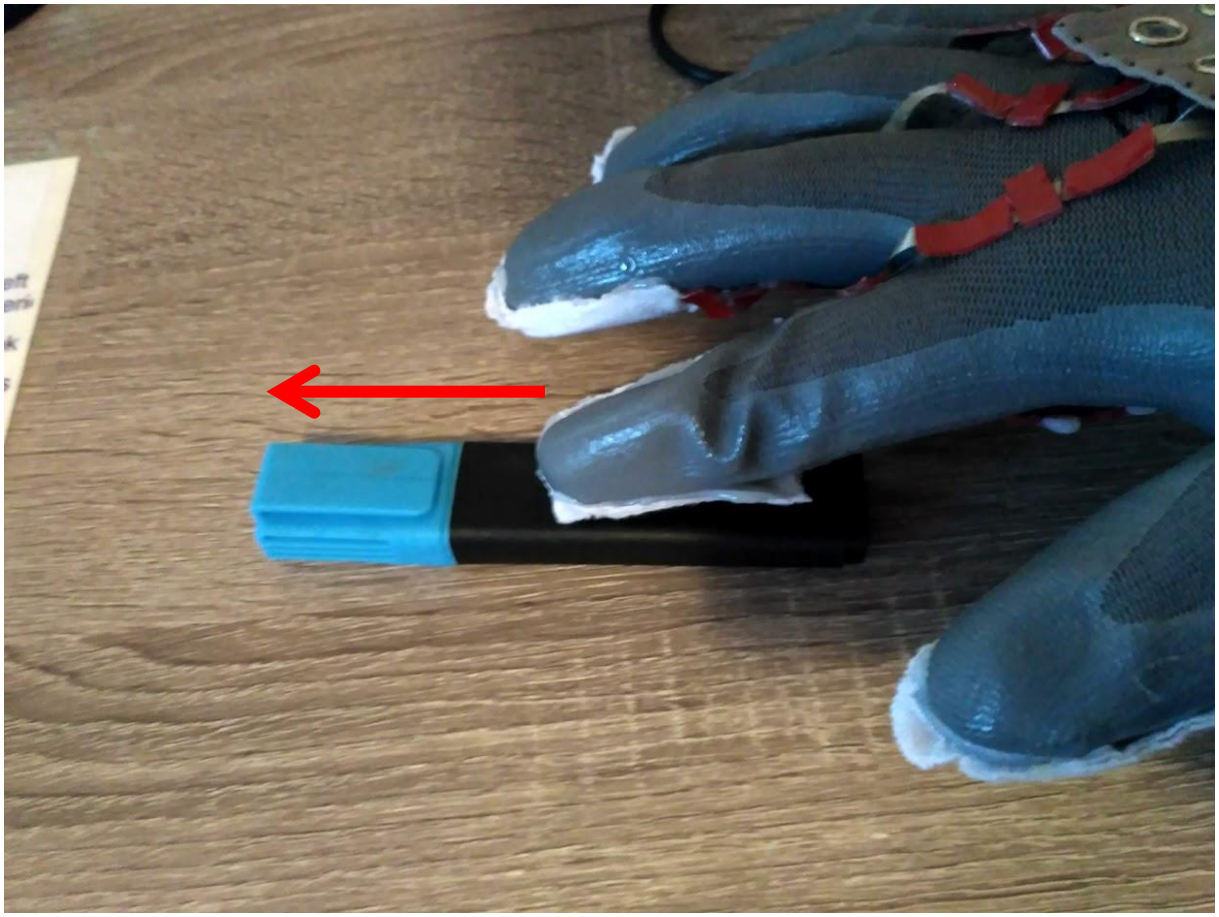
U svrhu vrednovanja ili čak moguće nadogradnje sustava MTM-2 s vrijednostima opterećenja na prstima pri rukovanju određenim predmetima, podatkovnom rukavicom snimljeno je nekoliko pokreta opisanih unutar sustava MTM-2. Značenje simbola i opis definiranih pokreta iz sustava MTM-2 iznijeti su u poglavlju:3.3.

7.1.1. Pokret uzimanja, slučaj GA

Za snimanje ovog pokreta korišten je flomaster duljine 11 cm i širine 2 cm (Slika 57.). Izmjerena masa korištenog predmeta iznosi 0,013 kg. Pokret GA izveden je kažiprstom kojim se ostvaruje pritisak flomastera o podlogu te istovremenim guranjem flomastera po tvrdoj podlozi. Ovim pokretom flomaster je pomaknut 15 cm po podlozi, a izvođenje pokreta traje oko 1 s.

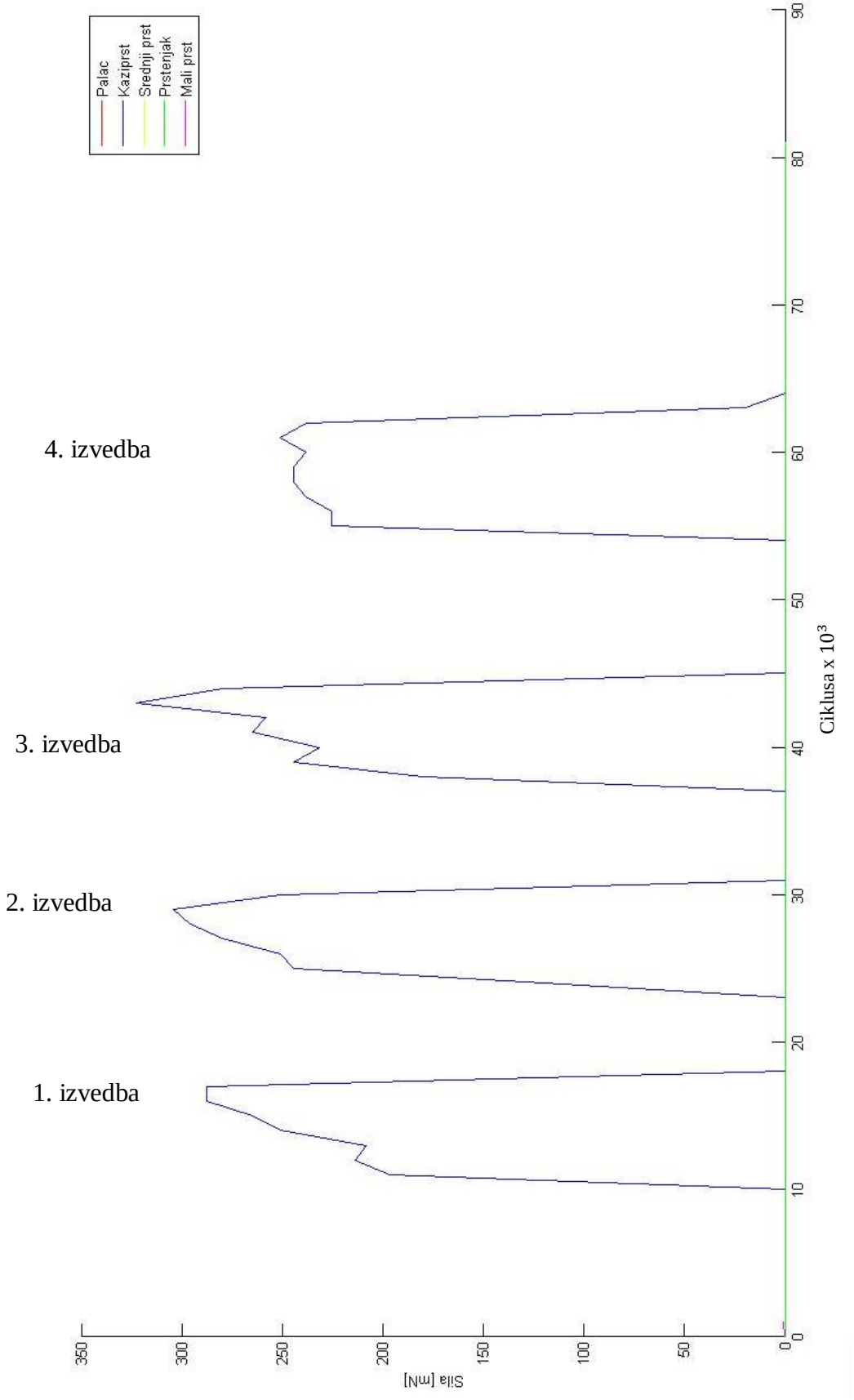


Slika 57. Flomaster na vagi



Slika 58. Izvedba pokreta GA

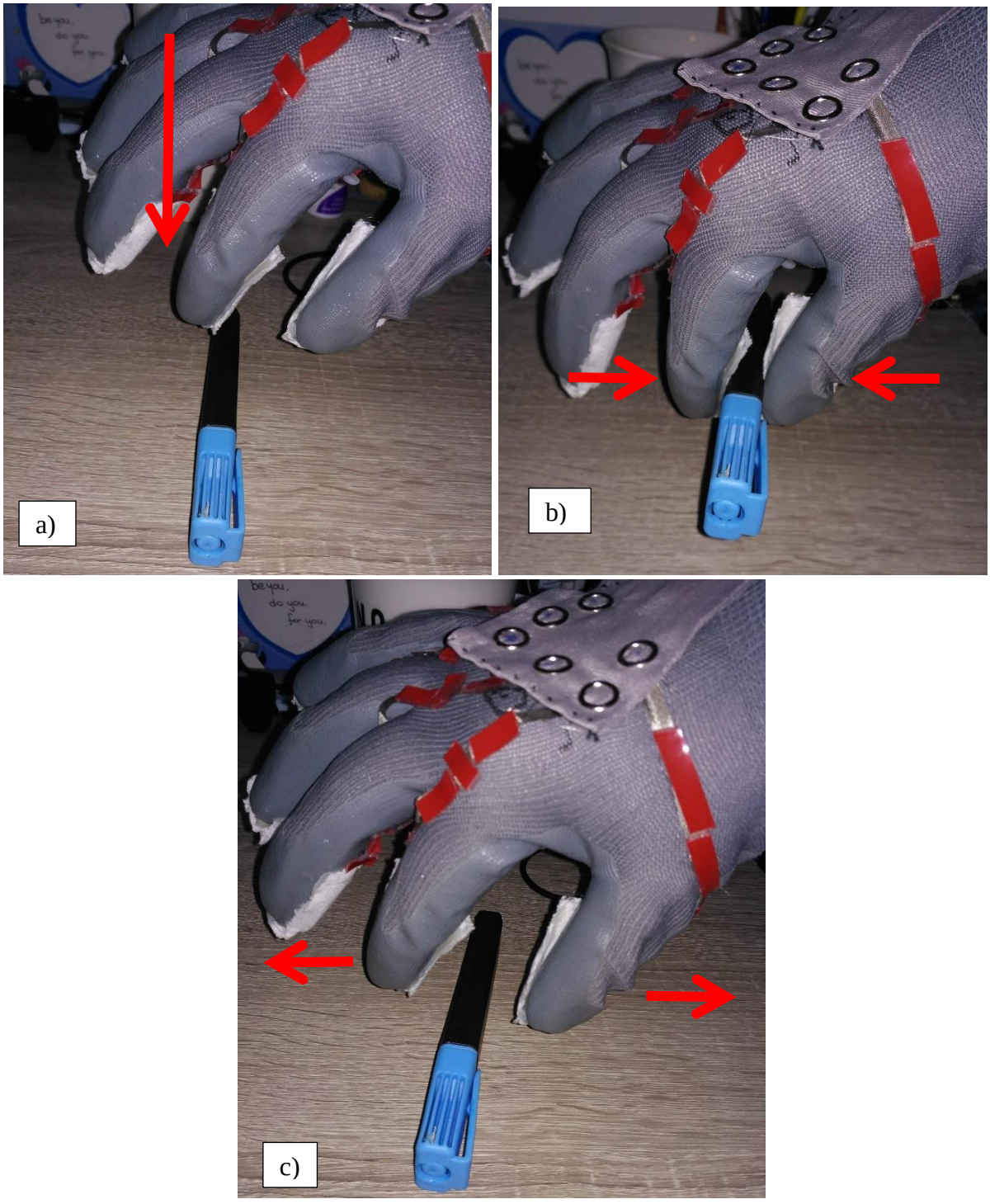
Ponašanje senzora za vrijeme ove radnje vidljivo je na slici 59. Pokret guranja flomastera izveden izveden je četiri puta te su zabilježene slične vrijednosti. Očitane vrijednosti normalne sile ostvarene pritiskom prsta, potrebne da se flomaster pomiče po stolu, kreću se između 0,25 i 0,3 N.



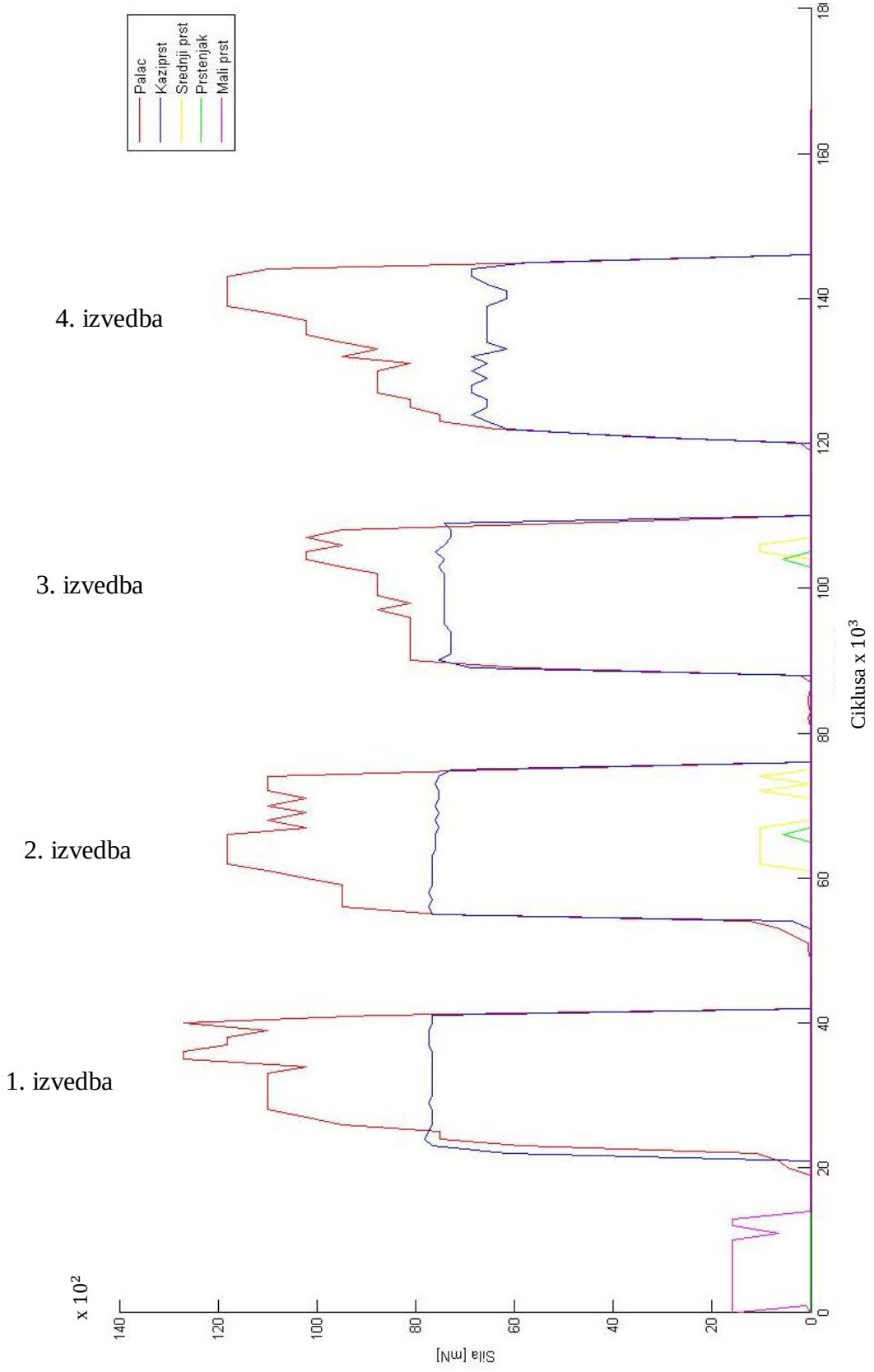
Slika 59. Snimljene vrijednosti za pokret GA

7.1.2. Pokret uzimanja, slučaj GB

U cilju snimanja pokreta GB korišten je isti predmet, flomaster, iz prethodnog primjera (7.1.1.). Pokret je raliziran hvatanjem predmeta palcem i kažiprstom, podizanjem na visinu 10 cm od podloge, držanjem u trajanju od 2 s te ispuštanjem predmeta (Slika 60.).



Slika 60. Pokret GB: a) posezanje, b) hvatanje, c) ispuštanje

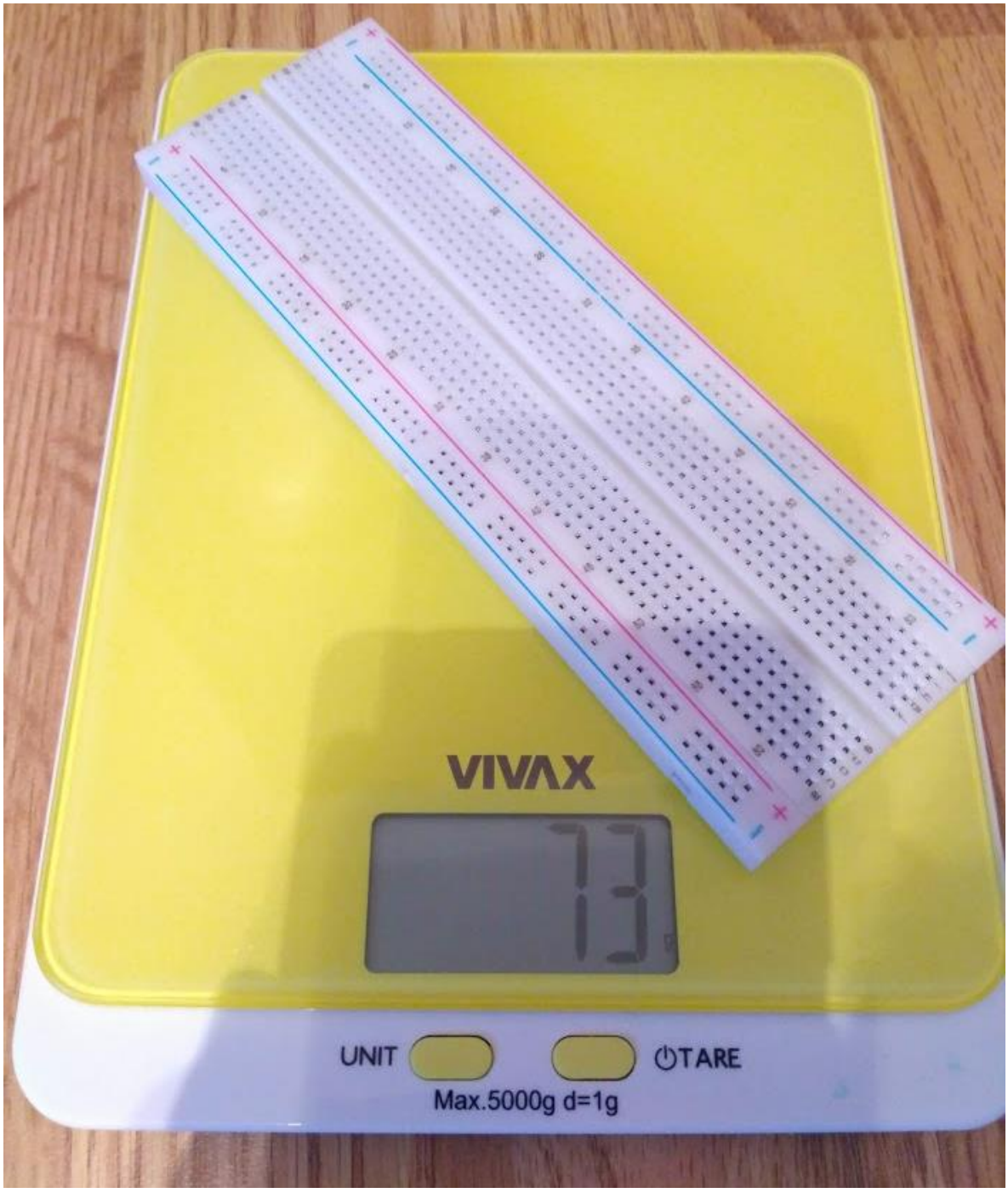


Slika 61. Snimljene vrijednosti za pokret GB

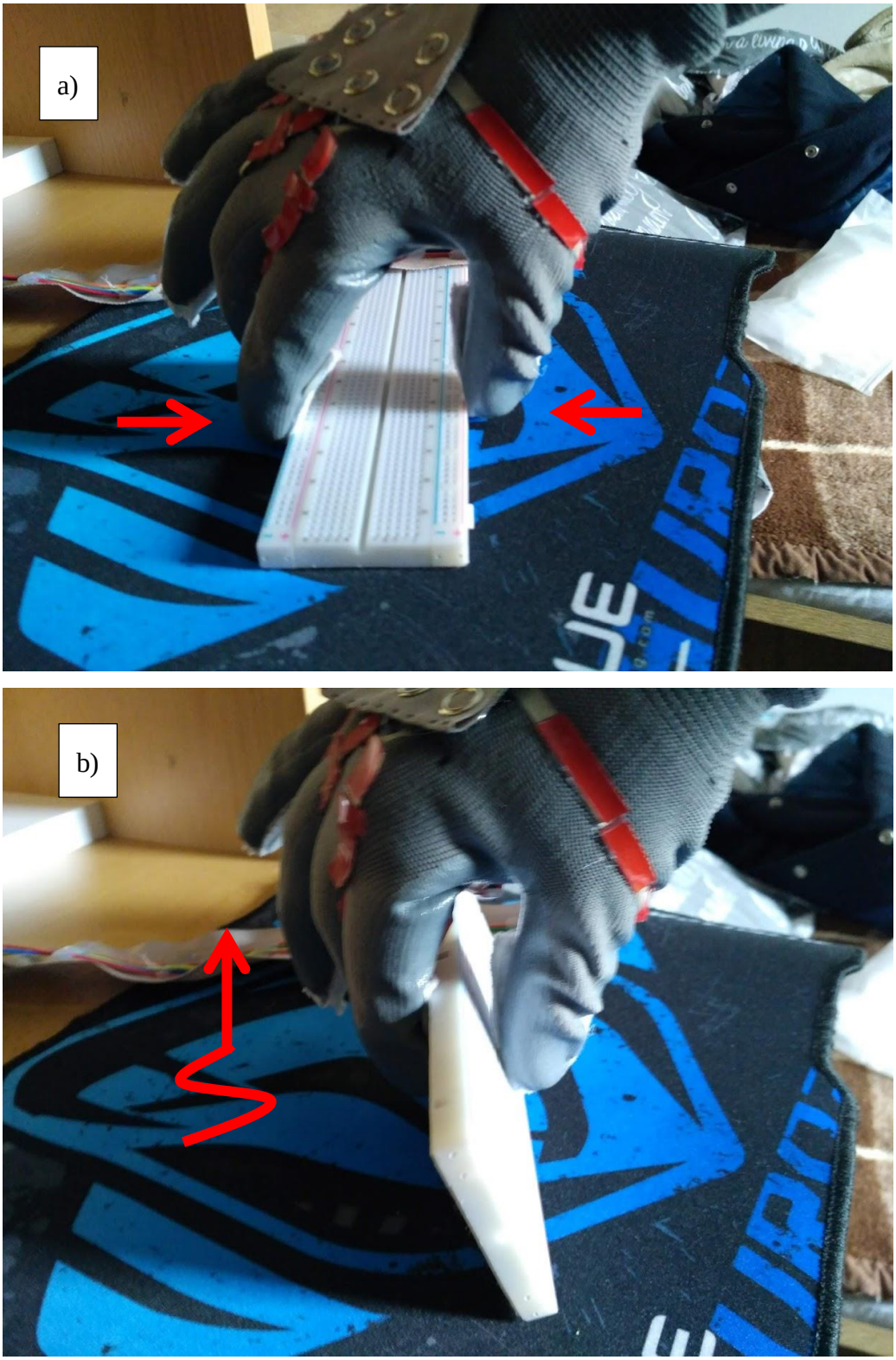
Iz grafa na slici 61. vidljivo je istovremeno opterećenje palca i kažiprsta. Hvatanjem flomastera stiskom palca i kažiprsta javljaju se reaktivne sile koje djeluju na ta dva prsta. Naime, zbog nesavršenosti izvedbe senzora, dodir na senzoru kažiprsta ostvaren je na većoj površini od senzora na palcu. Sukladno tome na palcu je snimljena veća vrijednost mase nego na kažiprstu. Ovaj rezultat ukazuje na to da će se bolja točnost postići korištenjem više manjih taktilnih senzora po svakom prstu što opet predstavlja izazov u fizičkoj izvedbi rukavice. Kod hvatanja flomastera zabilježene vrijednosti normalne sile iznose za palac od 0,8 do 1,2 N, a za kažiprst od 0,7 do 0,8 N.

7.1.3. Pokret uzimanja, slučaj GC

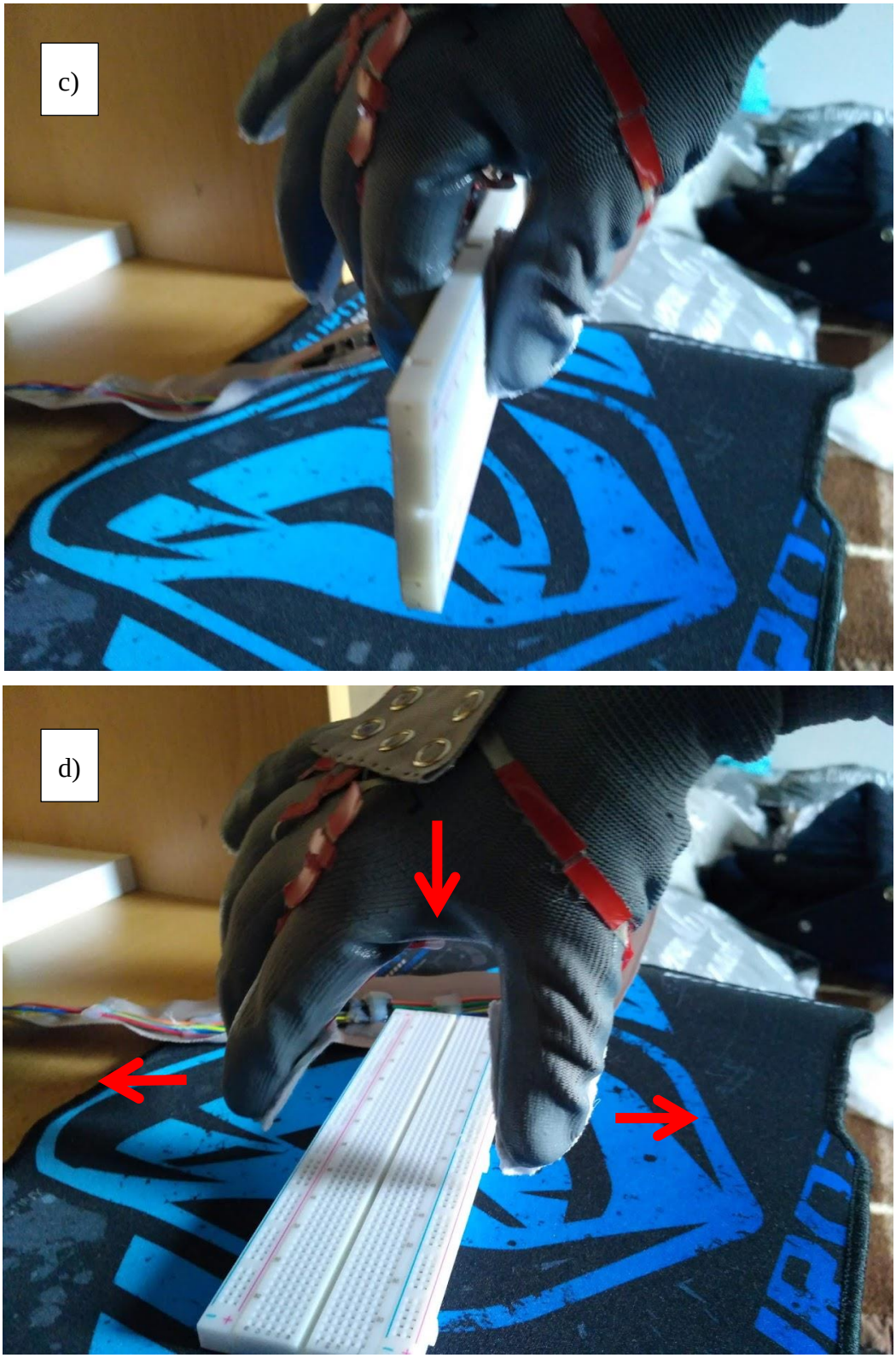
Za snimanje ovog pokreta korištena je pločica *breadboard* koja je u obliku kvadra, dimenzija 16x5,5x0,8 cm. Visina *breadboarda* znatno je manja u odnosu na ostale dimenzije (Slika 62.). S obzirom na njene dimenzije teško ju je podignuti jednim pokretom već ju je potrebno prstima prvo malo izdići te zatim uhvatiti. Pokret je izveden djelovanjem palca, kažiprsta i srednjeg prsta. Predmet je podignut 10 cm od podloge te držan na toj visini u trajanju od 4 s. Masa *breadboarda* je 0,073 kg.



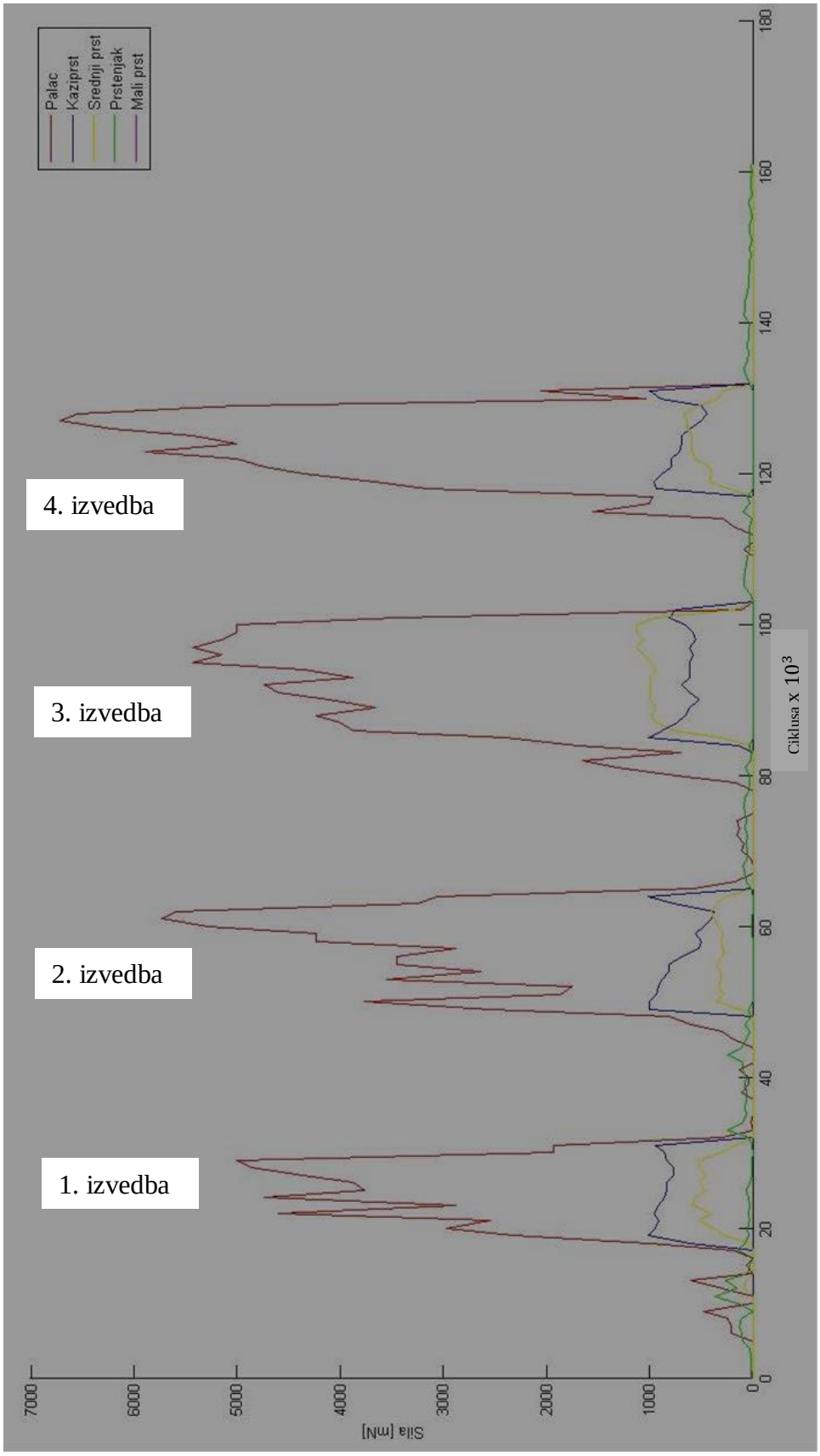
Slika 62. *Breadboard* na vagi



Slika 63. Pokret GC (prvi dio): a) posezanje, b) hvatanje



Slika 64. Pokret GC (drugi dio): c) držanje, d) ispuštanje



Slika 65. Snimljene vrijednosti za pokret GC

Prema slici 65. vidljivo je da su tri senzora aktivna. Najveća sila pojavljuje se na palcu dok se manje vrijednosti pojavljuju na kažiprstu i srednjem prstu.

U tablici 4. dana je usporedba snimljenih vrijednosti za tri slučaja uzimanja: GA, GB i GC. S obzirom na nesavršenosti izvedbe ove podatkovne rukavice, vrijednosti u tablici su okvirne, ali daju uvid u međusobni odnos opterećenja na pojedinim prstima pri određenim radnjama. Promatranjem tih odnosa moguće je razviti postupak zaključvanja o vrsti pokreta na temelju dodirnih sila. Naprimjer, za silu na kažiprstu iznosa 0,3 N koja je aktivna 1 s moguće je pretpostaviti da je riječ o guranju predmeta mase manje od 0.3 kg.

Tablica 4. Prosječne vrijednosti normalnih sila za snimljene pokrete

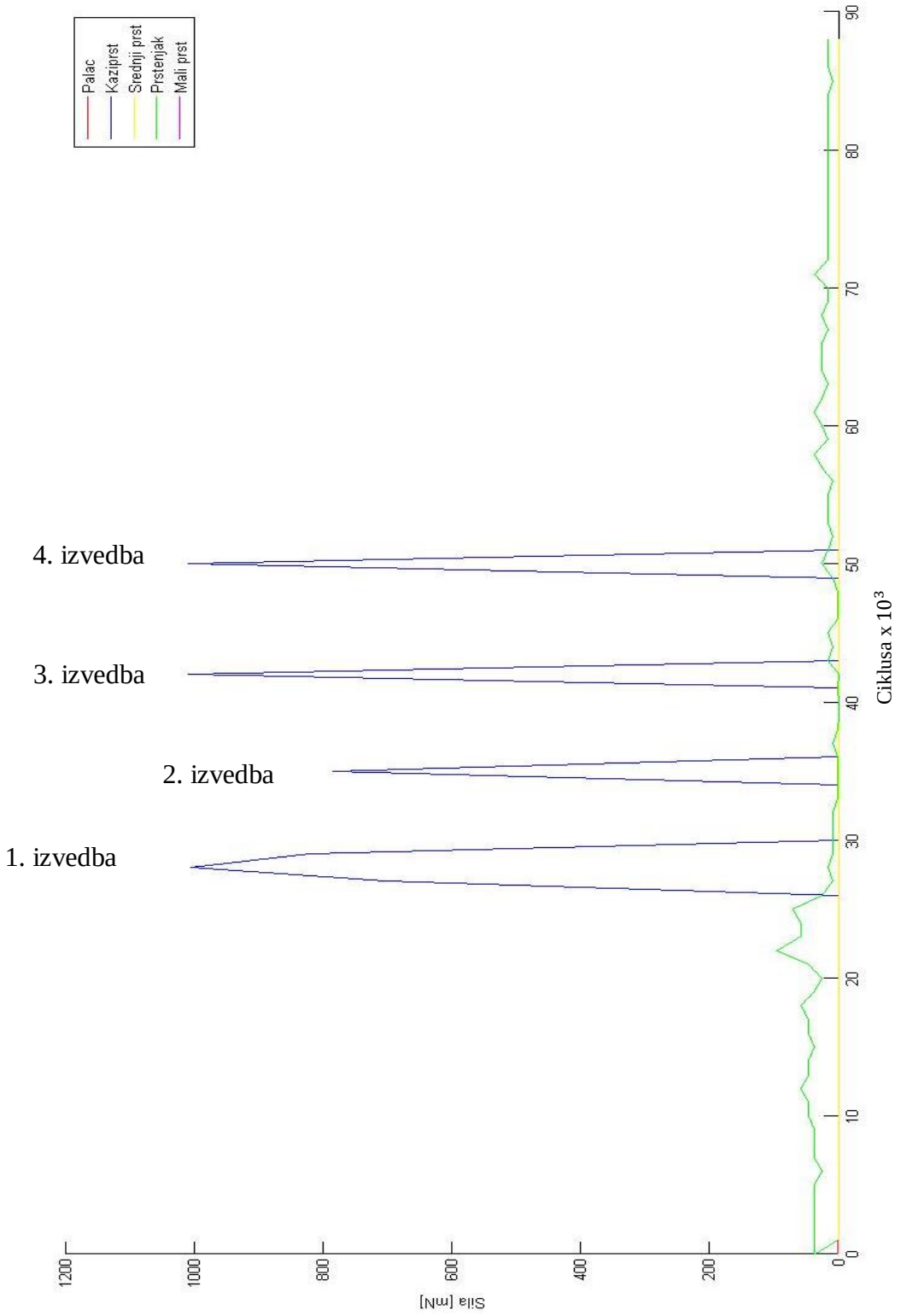
Pokret	Predmet	Masa predmeta [kg]	Vrijeme izvođenja [s]	Sila na palcu	Sila na kaži-prstu	Sila na srednjem prstu	Sila na prstenjaku	Sila na malom prstu
				[N]				
GA	Flomaster	0,13	1	0	0,3	0	0	0
GB			2	1,2	0,8	0	0	0
GC	*Breadboard*	0,73	4	6,7	1	0,5	0	0

7.2. Snimanje nekih učestalih pokreta

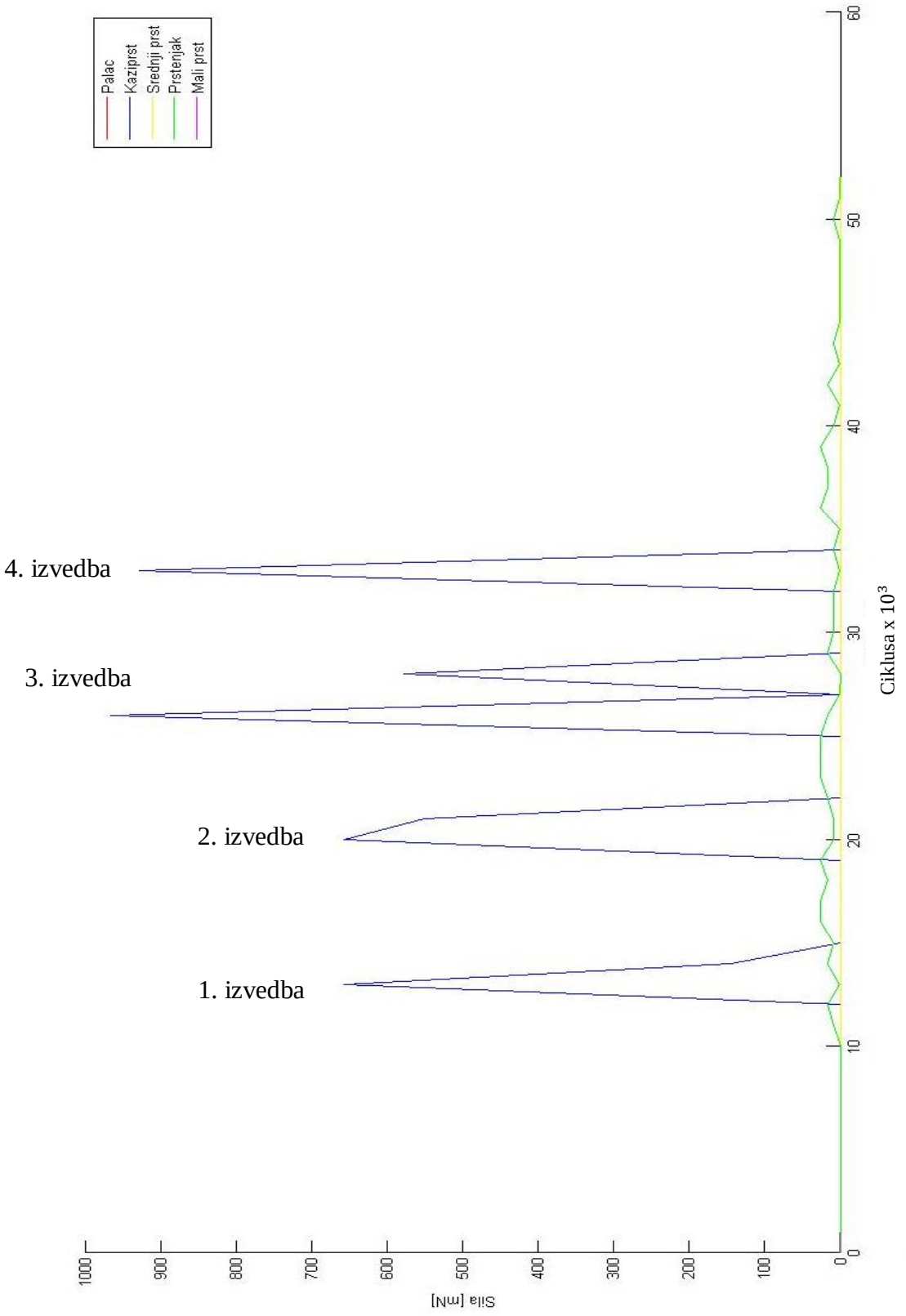
Ovdje su promatrana opterećenja kod nekih rutinskih pokreta rukama koje ljudi vrlo često izvode u svom svakodnevnom životu.

7.2.1. Klikanje na mišu

Podatkovnom rukavicom zabilježen je pokret kojim se ostvaruje klik na mišu. Trajanje jednog klika u prosjeku iznosi 0,3 s, dok kod dvoklika iznosi oko 0,7 s. U nastavku je prikaz ponašanja senzora za jedan klik (Slika 66.) i za dvoklik (Slika 67.). Izmjerena potrebna sila za ostvarivanjem klika na mišu iznosi u prosjeku 1 N. Vidljiva razlika između klika i dvoklika je da se kod dvoklika radnja obavi jako brzo te su vrijednosti stiska kažiprsta nešto manje nego kod jednog klika.U izvedbi jednog klika i nakon što je tipka na mišu svladana traje mali porast sile, odnosno, kažiprstom se ostvaruje nešto duži i intenzivniji stisak od dovoljnog.



Slika 66. Snimljene vrijednosti za jedan klik na mišu



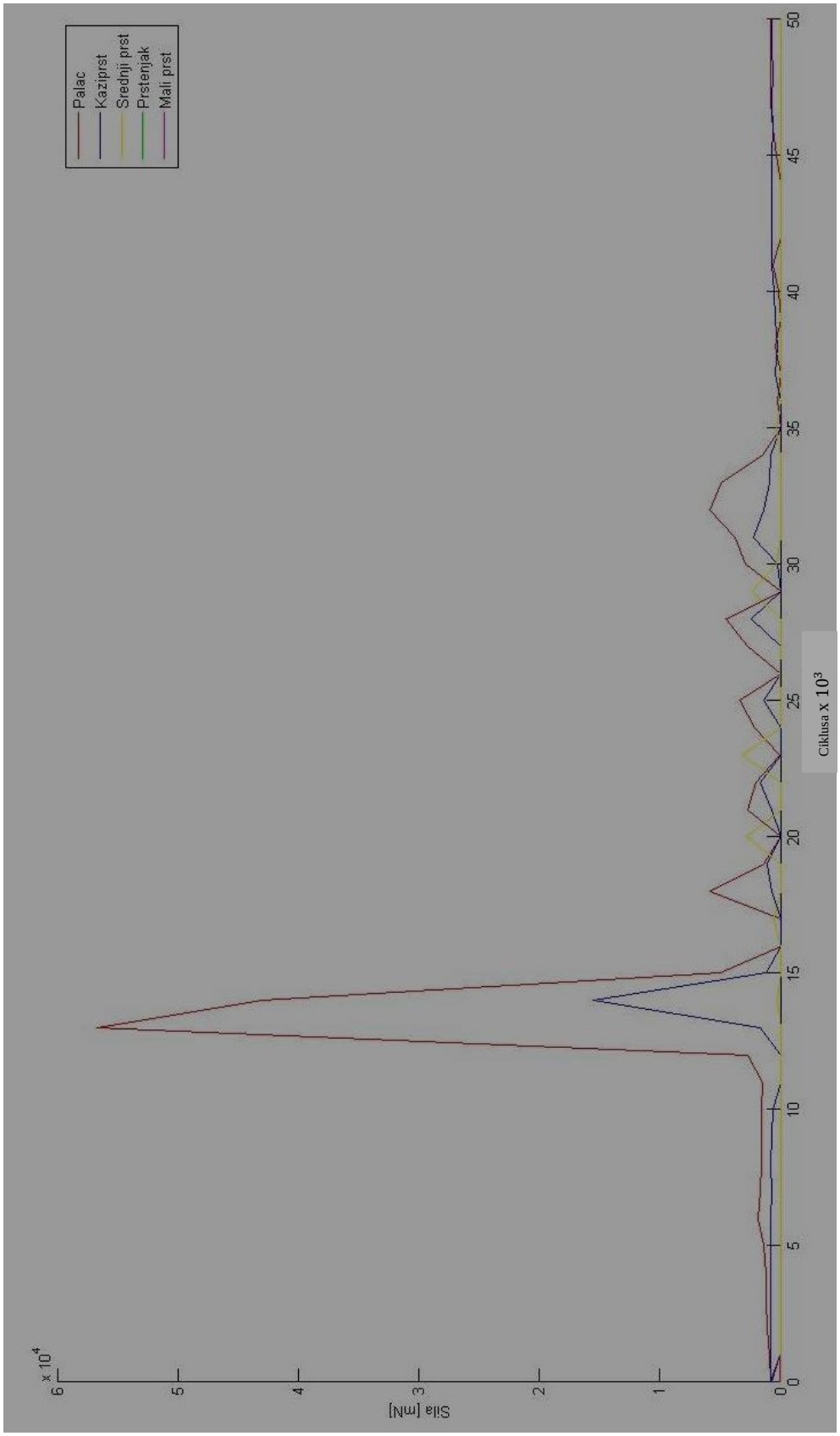
Slika 67. Snimljene vrijednosti za dvoklik na mišu

7.2.2. Uvrtanje i odvrtanje čepa boce

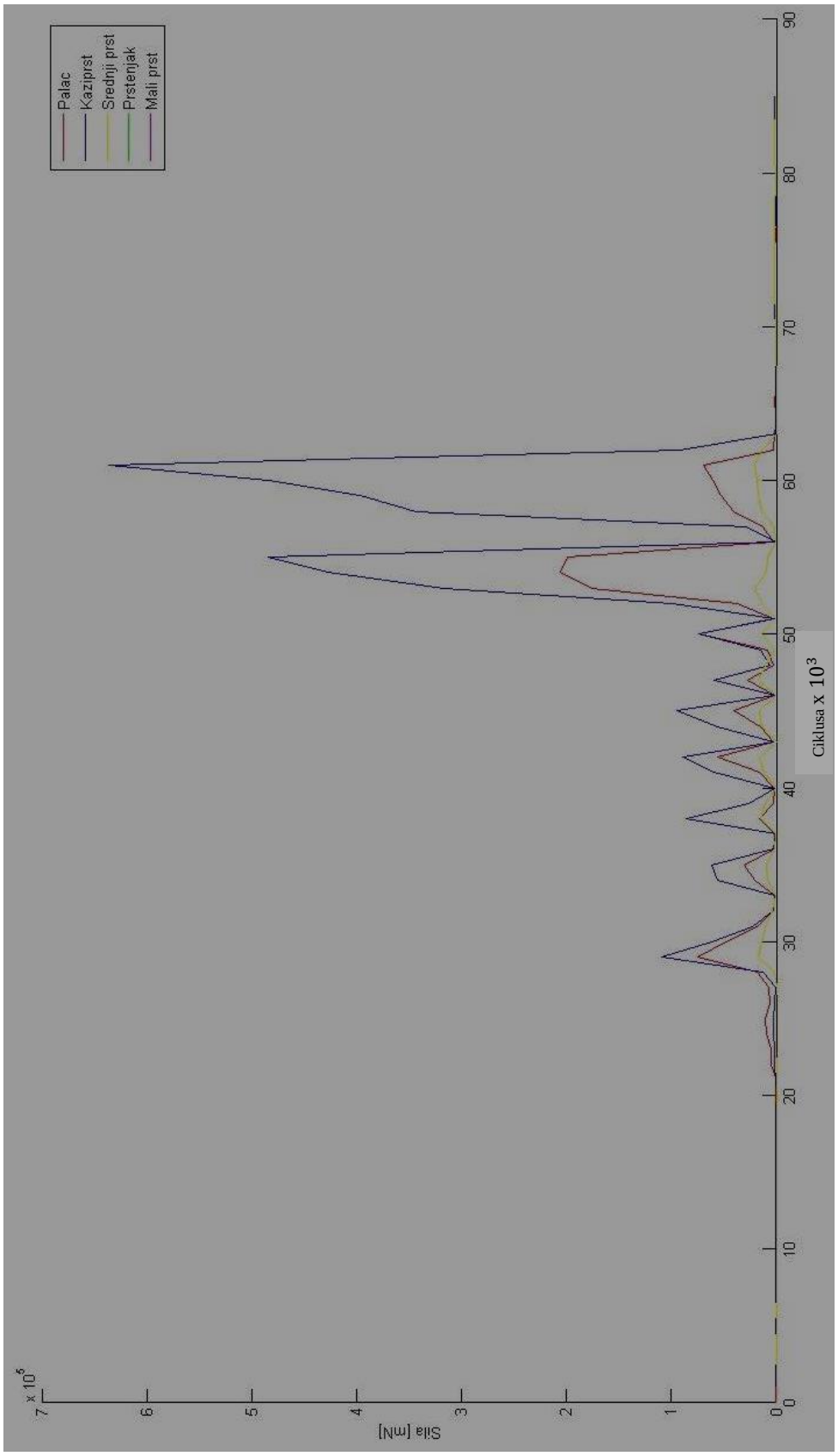
U okviru ovog eksperimenta provođeno je snimanje odvrtanje pritegnutog čepa boce, a zatim i uvrtanje te zatezanje čepa na boci. Za ovaj eksperiment korištena je boca na kojoj je čep već bio otvaran. U dodiru s čepom sudjeluju tri prsta, palac, kažiprst i srednji prst. Zanimljivo je vidjeti iz snimljenih vrijednosti (Slika 69., Slika 70.) kako je kod odvrtanja čepa u lijevu stranu veće opterećenje na palcu dok je kod uvrtanja to na kažiprstu. Također zbog zategnutog čepa u startu odvrtanja su prisutne veće sile dok se kod uvrtanja jače opterećenje pojavljuje na kraju zbog zatezanja.



Slika 68. Uvrtanje i odvrtanje čepa boce



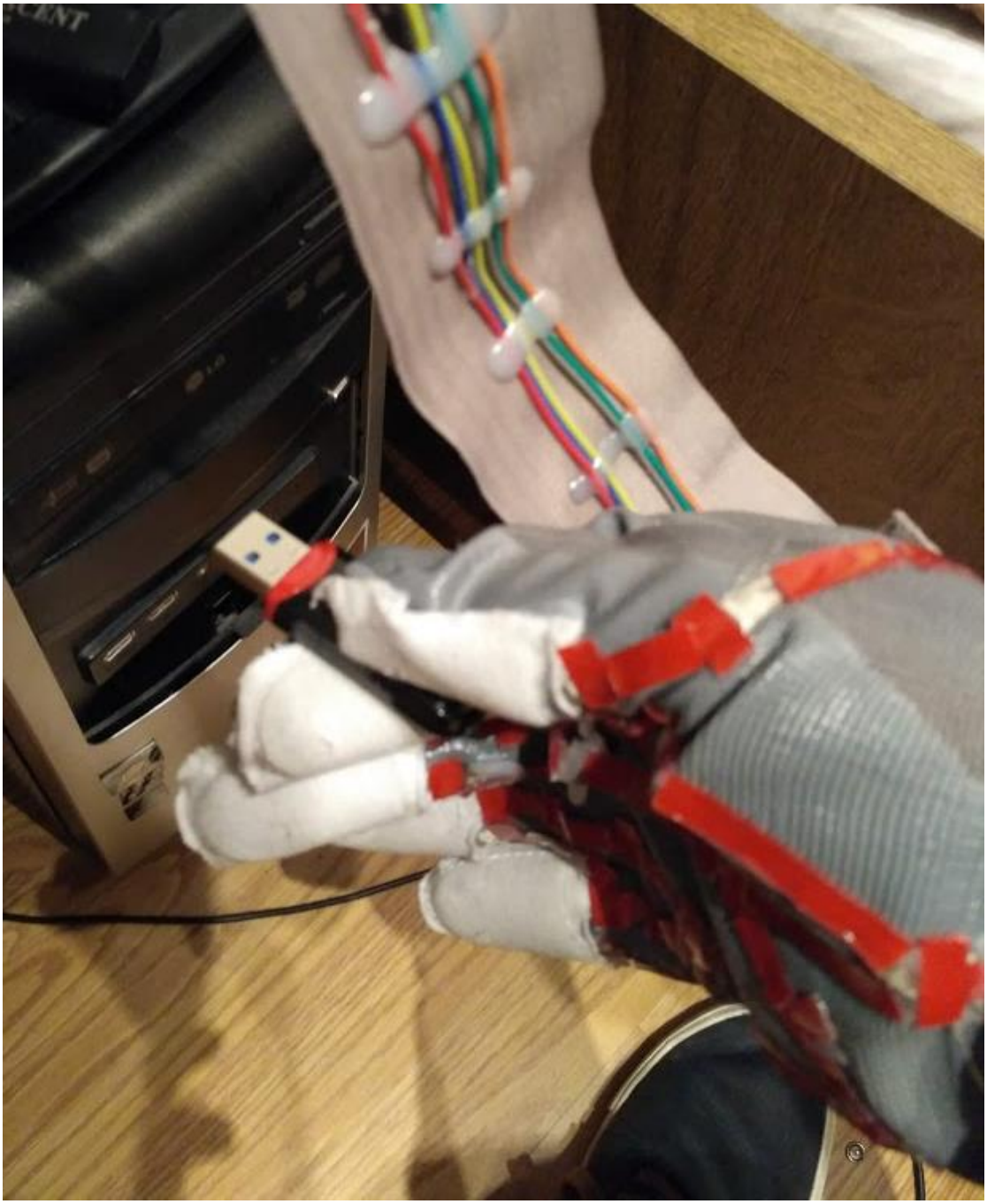
Slika 69. Odvrtanje zategnutog čepa boce



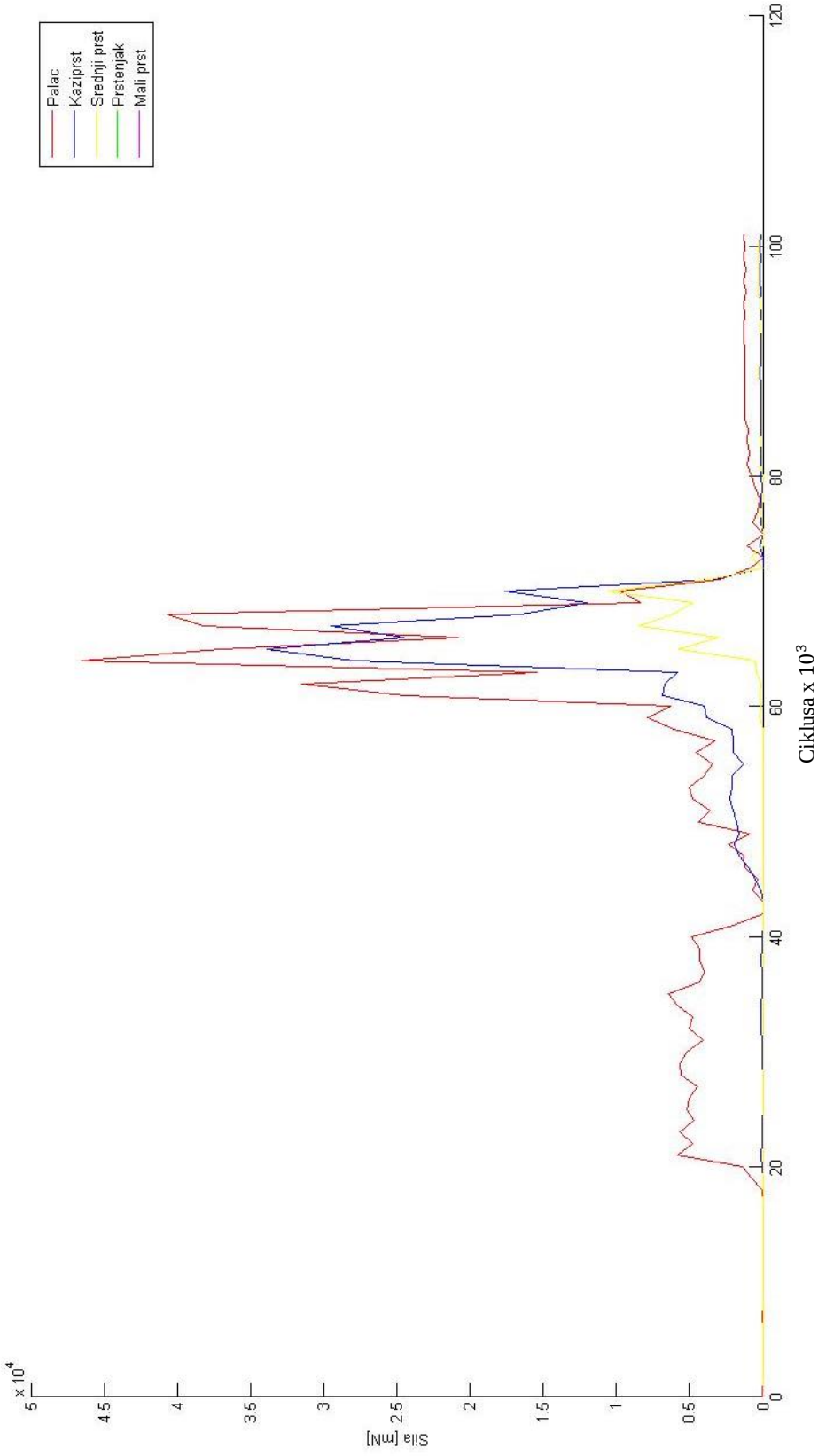
Slika 70. Uvrtanje i zatezanje čepa boce

7.2.3. Umetanje i vađenje USB *sticka*

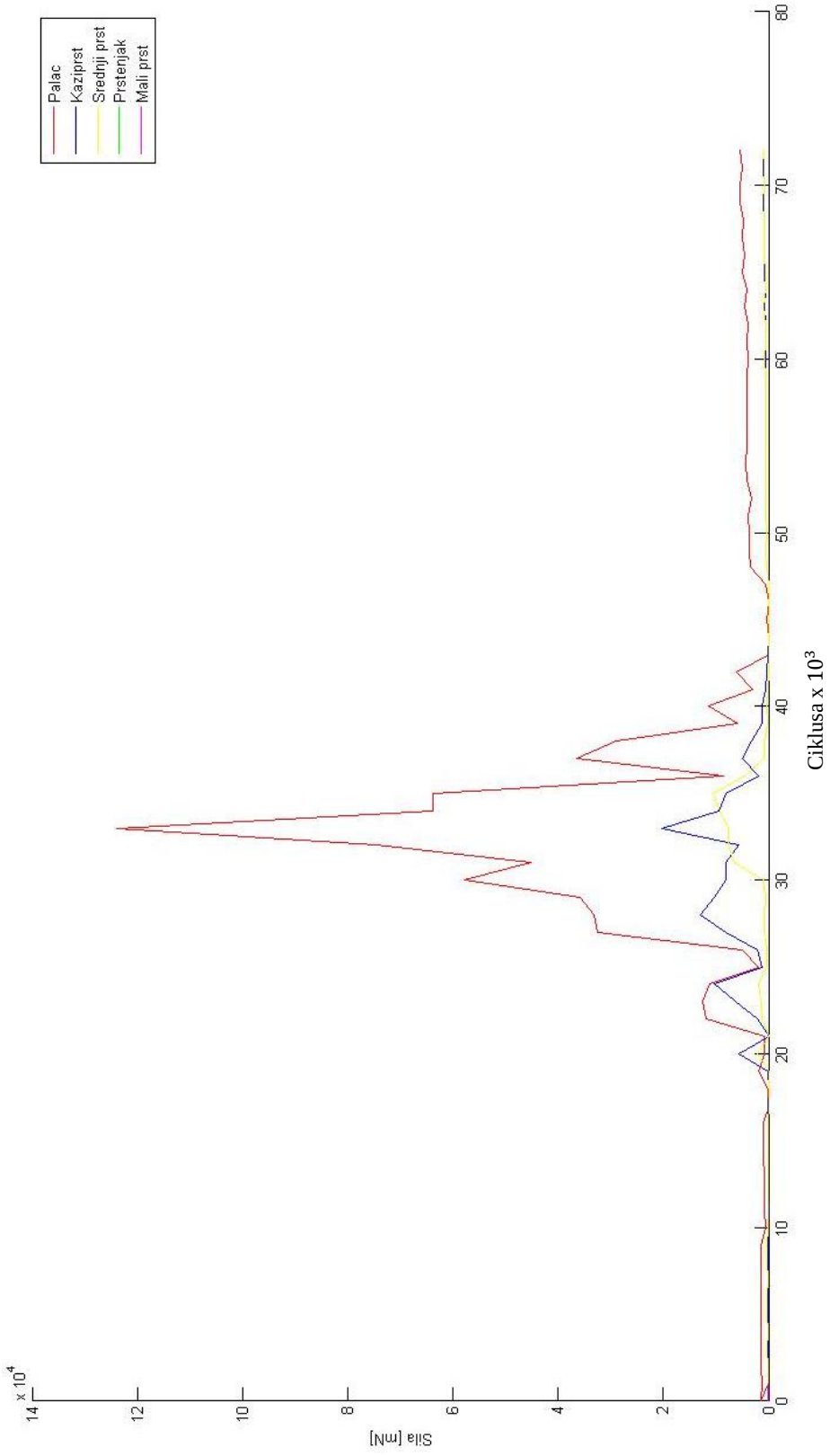
U nastavku je dan prikaz snimljenih vrijednosti za radnju umetanja i vađenja USB *sticka* u i iz USB sučelja na računalu (Slika 72., Slika 73.). USB *stick* je uhvaćen kažiprstom i palcem te umetnut i izvađen iz USB ulaza na računalu. Jedno izvođenje pokreta trajalo je dvije do tri sekunde.



Slika 71. Držanje USB *sticka*



Slika 72. Snimljene vrijednosti za umetanje USB sticka



Slika 73. Snimljene vrijednosti za vađenje USB *sticka*

7.3. Zaključivanje o rukovanom objektu

Daljnjim eksperimentiranjem promatrat će se rezultati senzora za rukovanje sličnim predmetima. Eksperiment će biti proveden držanjem prazne čaše vode. Potom će se isti slučaj snimiti i za čašu punu vode te za držanje dva valjkasta predmeta različitih promjera. Za sve slučajeve izvedena je ista kombinacija pokreta koja se sastoji od: posezanja, hvatanja, dizanja predmeta na visinu 15 cm od podloge, vraćanja predmeta u početni položaj te ispuštanja. Predmeti su u zraku držani u prosjeku 5 s, a za svaki predmet je ponovljeno mjerenje četiri puta.

7.3.1. Držanje prazne čaše vode

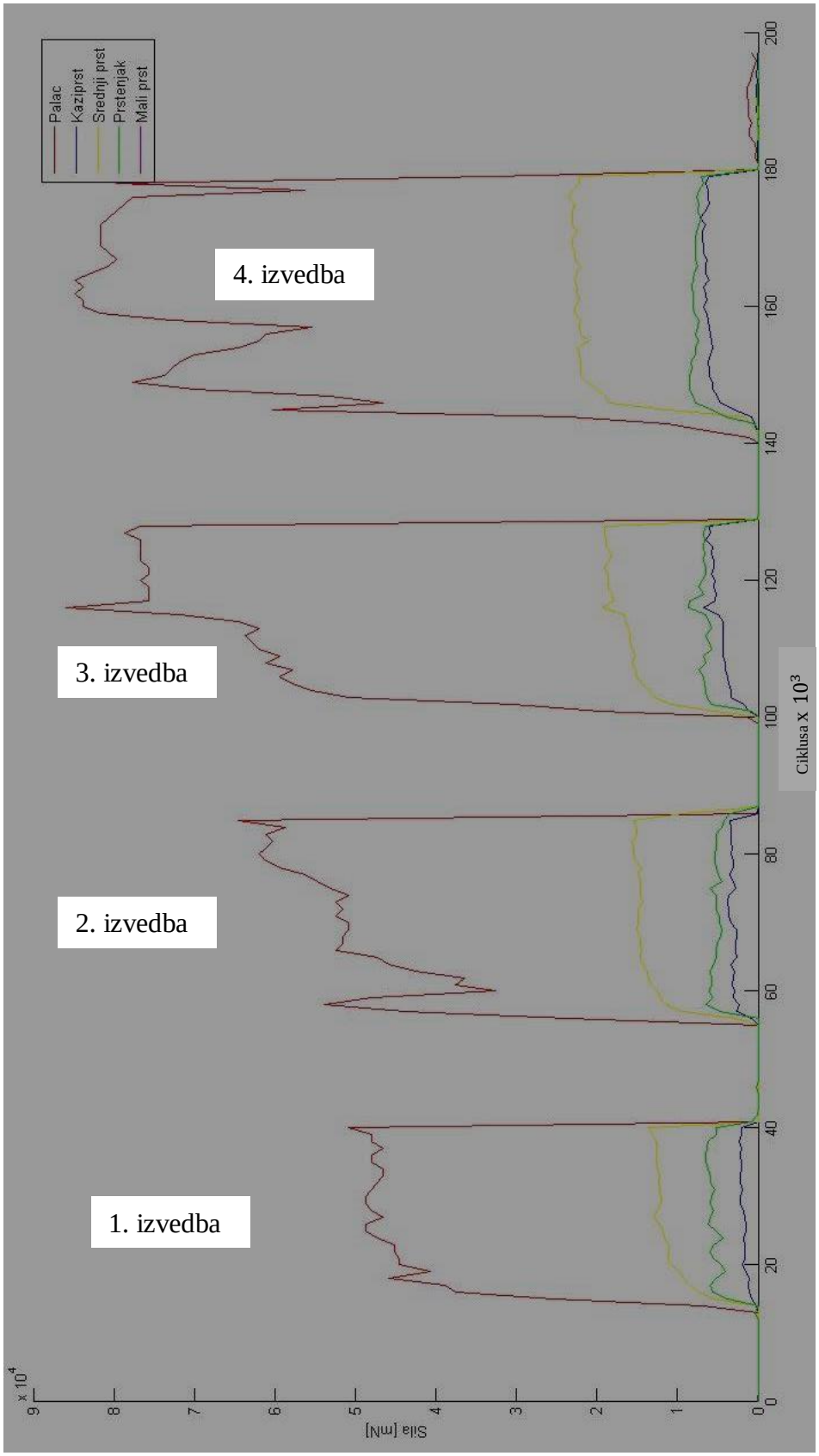
Prvi eksperiment proveden je držanjem prazne čaše čija je izmjerena masa 0,262 kg. Promjer čaše je 7 cm, a visina 12 cm.



Slika 74. Prazna čaša vode na vagi



Slika 75. Držanje čaše



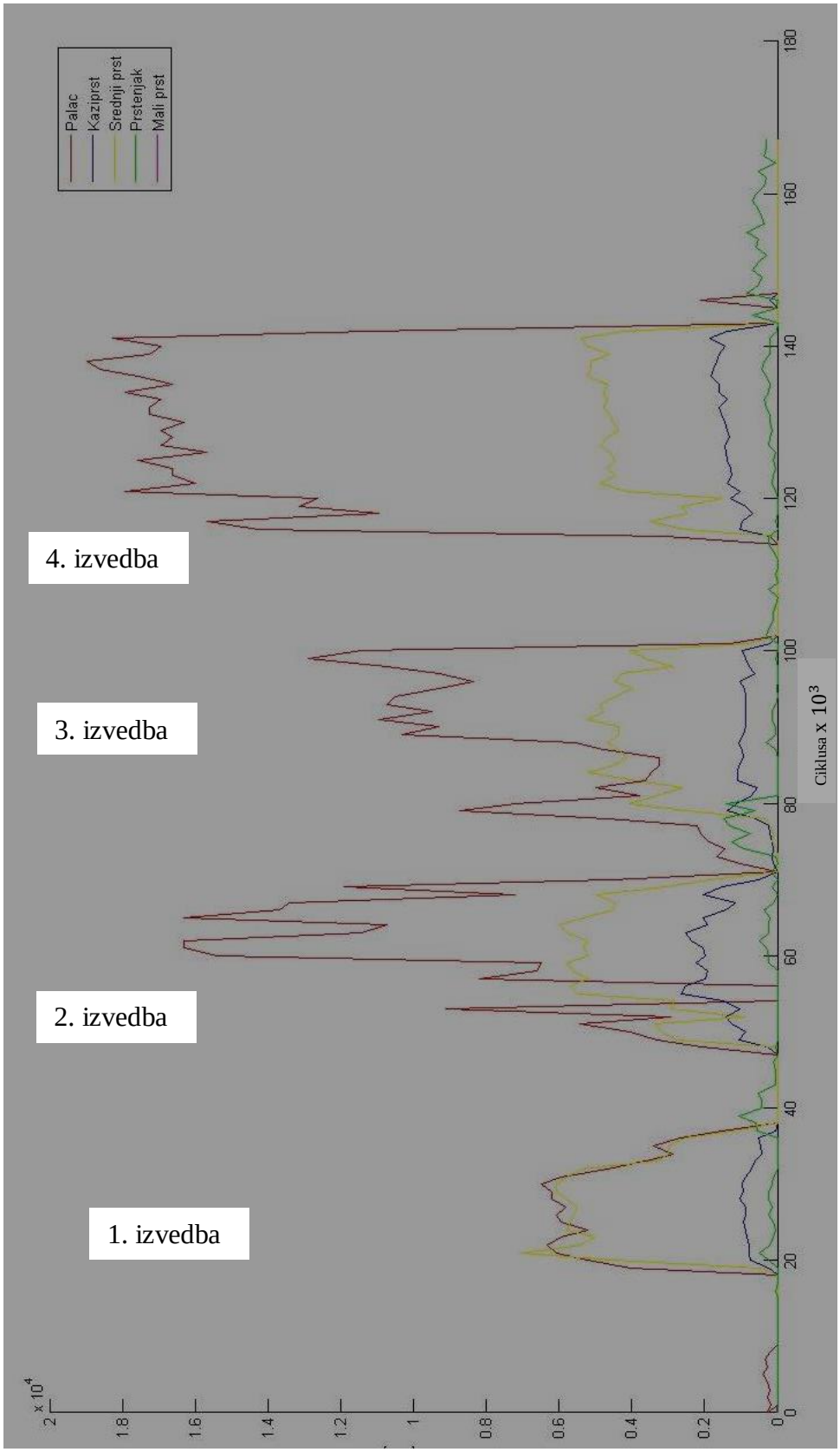
Slika 76. Snimljene vrijednosti za držanje prazne čaše vode

7.3.2. Držanje čaše ispunjene vodom

Slijedi eksperiment držanjem čaše ispunjene vodom čija je izmjerena masa 0,441 kg. Hvatanje, držanje i ispuštanje čaše realizirano jednako kao i u slučaju s praznom čašom (Slika 75.).



Slika 77. Čaša ispunjena vodom na vagi



Slika 78. Snimljene vrijednosti za čašu ispunjenu vodom

7.3.3. Držanje valjkastih predmeta

Provedena su još dva eksperimenta za valjkaste predmete različitih masa za koje je proveden isti eksperiment kao i s čašom.

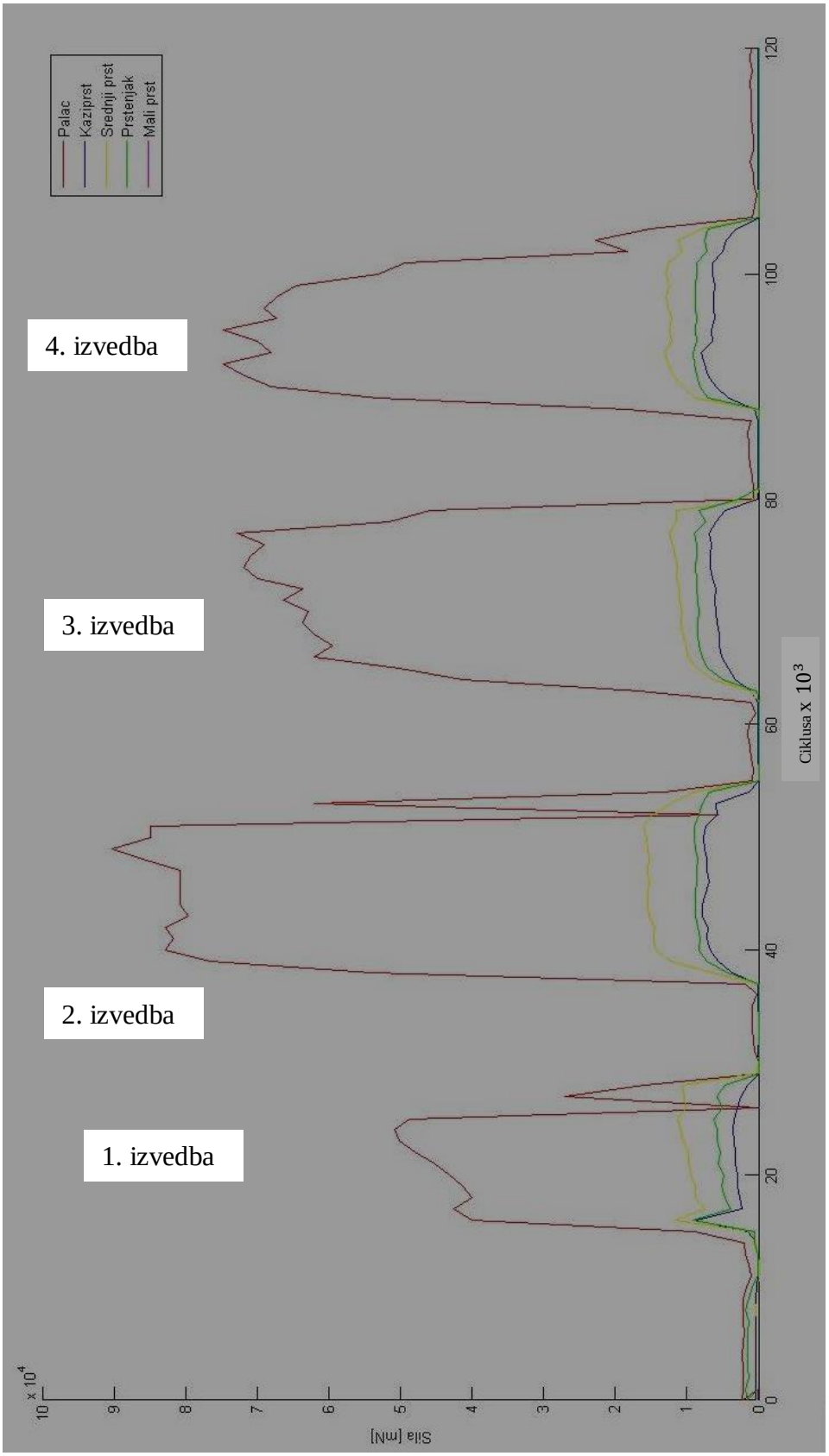
Prvi korišteni predmet je tuljac (Slika 79.) promjera 6 cm i visine 30 cm.



Slika 79. Tuljac na vagi



Slika 80. Držanje tuljca



Slika 81. Snimljene vrijednosti senzora za tuljac

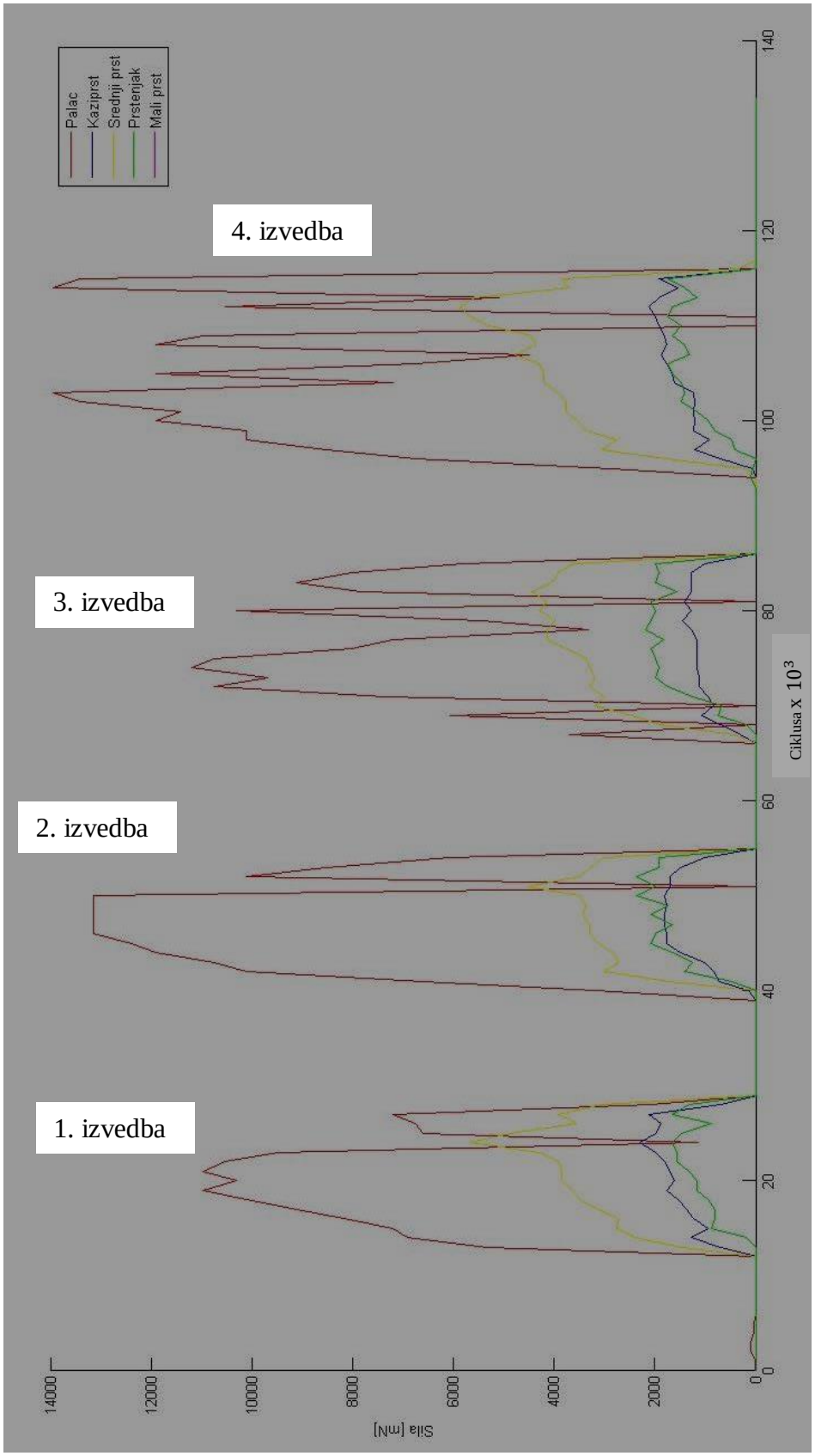
Drugi korišteni predmet je valjkasto kućište parfema promjera 5 cm i visine 25 cm.



Slika 82. Kućište parfema na vagi



Slika 83. Držanje kućišta parfema



Slika 84. Snimljene vrijednosti senzora za kućište parfema

Tablica 5. Prosječne snimljene vrijednosti za različite predmete

Predmet	Masa predmeta [kg]	Sila na palcu [N]	Sila na kažiprstu [N]	Sila na srednjem prstu [N]	Sila na prstenjaku [N]	Sila na malom prstu [N]
Prazna čaša	0,262	6	0,3	1,5	0,5	0
Čaša s vodom	0,441	16	2	6	1	0
Tuljac	0,201	75	7	12	10	0
Kućište parfema	0,366	13	2	2	4	0

Na temelju izmjerenih vrijednosti vidljiva je greška mjerenja. Kod hvatanja valjkastih predmeta za očekivati je da će opterećenje na palcu biti podjednako zbroju opterećenja na ostalim prstima. S obzirom da svi prsti kod hvatanja ne prianjaju jednakom površinom dobivene su različite vrijednosti. Palac u ovom slučaju ne obuhvaća stjenku čaše ili valjkastog predmeta punim jagodičnim dijelom prsta kao ostali aktivni prsti te se ne ostvaruje jednaka površina dodira. Ovakva zapažanja upućuju na to da je za točnije izmjere potrebno više senzora po pojedinom prstu.

Unatoč primijećenoj greški samih vrijednosti opterećenja, uočene su sličnosti izmjerenih opterećenja za pojedine predmete. S obzirom na ponavljanje sličnih odnosa opterećenja na pojedinim prstima u jednakom vremenskom trajanju zaključeno je da je u sklopu budućeg rada na temelju izmjerenih vrijednosti sila na pojedinim prstima moguće razviti algoritam za zaključivanje o obliku ili masi predmeta. Ovakvim pristupom moguće je robotima na temelju dodira omogućiti percepciju o okolišu u kojem se nalazi.

Također je zapaženo „poskakivanje“ vrijednosti na pojedinim senzorima, naročito kod malog promjera predmeta što upućuje na to da stisak čaše prstima nije konstantan. Pri držanju predmeta u zraku, primijećeno je da ga se ne drži konstantnom intenzivnošću, naime, uz istodobno pokušavanje fotografiranja izvođenog pokreta, stisak prstiju se mijenjao. Unatoč izvjesnoj smetenosti, ljudski mozak vrlo dobro regulira potreban stisak da čaša ne ispadne iz ruke.

8. ZAKLJUČAK

Potreba za efikasnom izvedbom procesa zahtijeva stalno proučavanje i usavršavanje ljudskog tjelesnog rada, pa tako i pokreta ruku, bilo da se radi o proizvodnji, medicini, sportu, umjetnosti ili nekim svakodnevnim radnjama. U proizvodnji je težnja razviti što brži i efikasniji način rada, u sportu je potrebno razviti vještinu kojom će se nadmašiti postojeći rekordi, u kirurgiji je potrebno optimirati i normirati kretnje kirurga kako bi se, uz ostalo, izbjegle greške. S obzirom na navedene potrebe, razvijani su, i još uvijek se razvijaju, sustavi za snimanje, oblikovanje i analizu rada, kao što je to naprimjer sustav MTM, kojima se ljudski rad raščlanjuje na pokrete i opisuje, s ciljem da se pokreti unaprijede, usvoje te uvježbaju.

U sustavu MTM opisani su elementarni tjelesni pokreti uključujući i vremena njihovog izvođenja. No, u sustavu MTM ne postoje egzaktne informacije o silama koje se pojavljuju pri pokretima ruku, a što je od iznimnog značaja za razvijanje što bolje metode rada, ali i s važnim utjecajem na ergonomsko oblikovanje alata i proizvoda.

Za nadogradnju sustava MTM, gdje bi osim informacija o trajanju nekog pokreta bile dostupne i informacije o iznosima sila pri pokretima, u ovome je radu razvijena podatkovna rukavicu koja bi takve sile mogla snimiti. U današnje vrijeme razvijene su brojne podatkovne rukavice kojima se uglavnom snimaju pozicije ljudske ruke što se iskorištava u svrhe upravljanja računalom ili nekim uređajem te za prijenos informacija o poziciji i orijentaciji ruke u virtualnu stvarnost.

U ovom radu razvijena i ručno izrađena podatkovna rukavica sadrži tekstilne i taktilne senzore osjetljive na sile kojima ih se opterećuju. Senzori se sastoje od elektrovodljivog tekstila te velostata koji ima svojstva da mijenja električni otpor u ovisnosti o intenzivnosti dodirne sile. Mjerenjem električnih svojstava dobivena je slika o intenzivnosti opterećenja pa su ovakvi senzori primjenjivi na podatkovnoj rukavici. Smještanjem senzora na rukavicu, i njihovim povezivanjem s eletroničkim komponentama (Arduino Uno i ?) te razvijanjem

programa (programsko okružje KOJE?), realizirana je podatkovna rukavica. Rukavica je realizirana komponentama niske cijene (okvirna je cijena rukavice ? kn).

Podatkovnom rukavicom izvedeno je nekoliko eksperimenata, ne bi li se dobile informacije o opterećenjima na prstima šake pri izvođenju nekih pokreta odnosno rukovanju određenim objektima. Pravednim eksperimentima, dijelom zasnovanih na sistematizaciji pokreta sustava MTM-2, dobivene su okvirne vrijednosti o silama na pojedinim prstima pri obavljanje određenih pokreta, naprimjer, u eksperimentu ? ustanovljeno je da postoji sila iznos ? N.

Eksperimentiranjem je također zapaženo i da se kod rukovanja raznim objektima vrlo često koristi bočni dio prstiju. Vrlo rijetko pokreti se, kao što je to naprimjer klik na mišu, obavljaju samo jagodičnim dijelom prsta. Ipak, i ovakav pokret u nekim slučajevima biva obavljen bočnim dijelom prsta. S obzirom na to, za snimanje opterećenja na prstima potrebno je više taktilnih senzora, a manjih površina, po svakom prstu kako bi se smanjila greška izmjerenih vrijednosti.

Snimanjem rukovanja predmetima koji su slični po svojim oblicima odnosno dimenzijama zapažene su slične vrijednosti na senzorima, što otvara put prema razvijanju algoritma o zaključivanju o obliku predmeta na temelju izmjerenih vrijednosti opterećenja koja djeluju na pojedini prst.

U svrhu razvijanja sljedeće, naprednije i još upotrebljivije podatkovne rukavice, potrebno je težiti prema senzorskom pokrivanju što većeg dijela šake i prstiju. U skladu s time, potrebno je razviti što manje senzore koji će jedni do drugih pokrivati cijelu površinu šake te davati kompletnu taktilnu sliku ruke s iznosima opterećenja na pojedinim mjestima. U takvim nastojanjima senzore je potrebno izraditi u manjim dimenzijama i preciznije, što je vrlo teško izvesti ručnim radom. Povećanjem broja senzora povećava se i broj električnih vodiča i spojeva na rukavicu, što također iziskuje njihovu što manju i precizniju izvedbu.

Osim mjerenja sile dodira, bitna komponenta u okviru određivanja kvalitete nekog rada je i pomak koji se ostvaruje rukom. Rukavicu je stoga potrebno nadograditi i sustavom za mjerenje pozicije unutar nekog radnog prostora.

Daljnjim usavršavanjem podatkovne rukavice u svrhu dobivanja točnijih podataka, moguće je rukovanjem određenim predmetima uvidjeti njihove ergonomske prednosti ili mane, ili pak

određenu metodu rada proglasiti više ili manje zahtjevnom od druge (kvantificirano vrednovanje metoda rada).

Također, posebno je atraktivna mogućnost istraživanja korištenja podatkovne rukavice pri virtualnom projektiranju radnih procesa, gdje se ona povezuje s postojećim inženjerskim CAD/CAE/CAM softverima.

9. LITERATURA

[1] https://www.tmphysio.com/resources/phases-of-throwing/ Pristupljeno: 20181208

[2] http://www.economicsdiscussion.net/engineering-economics/motion-study-meaning-objectives-and-procedure/21683 Pristupljeno: 20181208.

[3] Marko Krznar, MTM sistem za analizu ručnih i montažnih radova u serijskoj proizvodnji, Rade Končar 1986

[4] Zoran Kunica, Zavarivanje i montaža, nastavni materijal, FSB, Zagreb, 2019.

[5] https://en.wikipedia.org/wiki/Wired_glove Pristupljeno: 20190104.

[6] https://whatis.techtarget.com/definition/data-glove Pristupljeno: 20190104.

[7] http://www.historyofinformation.com/detail.php?entryid=4080 Pristupljeno: 20190104.

[7] http://www.historyofinformation.com/detail.php?entryid=4081 Pristupljeno: 20190105.

[8] https://en.wikipedia.org/wiki/Power_Glove Pristupljeno: 20190105.

[9] http://www.cyberglovesystems.com/contact/ Pristupljeno: 20190106.

[9] https://www.virtalis.com/blogs/products/cyberglove-systems/ Pristupljeno: 20190106.

[10] http://www.mellottsvrpage.com/index.php/the-p5-data-glove-review/ Pristupljeno: 20190109.

[11] https://ni.www.techfak.uni-bielefeld.de/node/3545 Pristupljeno: 20190110.

[13] http://www2.kelm.ftn.uns.ac.rs/literatura/si/pdf/senzori_u_robotici/sur4.pdf Pristupljeno: 20190114.

[14] https://e-radionica.com/hr/blog/2015/10/08/sto-je-arduino-i-croduino/ Pristupljeno 20190217.

[15] https://www.arduino.cc/reference/en/language/functions/analog-io/analogread/ Pristupljeno 20190218

[17] https://en.wikipedia.org/wiki/Thomas_A._DeFanti Pristupljeno 20190220

[18] https://en.wikipedia.org/wiki/Daniel_J._Sandin Pristupljeno 20190220

[19] https://www.evl.uic.edu/entry.php?id=2162 Pristupljeno 20190220

[20] http://www.w2vr.com/archives/Fisher/06_Dataglove.html Pristupljeno 20190221

[21] https://newatlas.com/go/1148/ Pristupljeno 20190222

[22] http://arduinotogo.com/2017/02/28/chapter-6-analog-input/ Pristupljeno 20190222

[23] https://www.nap.edu/read/4761/chapter/9 Pristupljeno 20190228

[24] https://repozitorij.unin.hr/islandora/object/unin:1177/preview Pristupljeno 20190302

[25] http://automatizacija1.etf.rs/poglavlja/Robotika%206.htm Pristupljeno 20190302

[26] https://repozitorij.etfos.hr/islandora/object/etfos:1496/preview Pristupljeno 20190302

[27] http://www.plc-bh.com/proizvodi/enkoderi/ Pristupljeno 20190302

[28] https://automatizacijairobotika.wordpress.com/2011/11/17/senzori-u-robotici/ Pristupljeno 20190302

[29] https://books.google.hr/books?id=5D3ZPW1lw6sC&pg=PA149&lpg=PA149&dq=elektromagnetski+senzori+polo%C5%BEaja&source=bl&ots=VMZdMoBEH5&sig=ACfU3U2OHmIoejS5iDuO6TuCqB2Yv0ivsQ&hl=hr&sa=X&ved=2ahUKEwirxser-OLgAhVBxMQBHbRaDjYQ6AEwBXoECAgQAQ#v=onepage&q=elektromagnetski%20senzori%20polo%C5%BEaja&f=false Pristupljeno 20190302

[30] https://www.racunalo.com/senzorske-tehnologije-u-smartphoneima-sto-trebate-znati-o-ziroskopu/ Pristupljeno 20190302

[31] https://learn.sparkfun.com/tutorials/flex-sensor-hookup-guide/all Pristupljeno 20190302

[32] http://www2.kelm.ftn.uns.ac.rs/literatura/si/pdf/senzori_u_robotici/sur4.pdf Pristupljeno 20190109

[33] http://automatizacija1.etf.rs/poglavlja/Robotika%206.htm

10.PRILOG

10.1. Programski kôd u programu IDE

10.1.1. Očitavanje i ispis vrijednosti senzora

```
//Definicija varijabli:
int a;
int b;
int c;
int d;
int e;

void setup() {
  //Ovaj dio koda čita se samo jednom
  //Pokretanje serijske komunikacije računala i pločice brzinom od 9600 bita u sekundi
  Serial.begin(9600);
}

void loop() {
  //Ovaj dio koda se ponavlja
  a=analogRead(A0); //Čitanje vrijednosti napona s analognog pina A0 i dodjeljivanje vrijednosti varijabli a
  delay(50);        //Stanka od 50 ms prije nastavka
  b=analogRead(A1); //Čitanje vrijednosti napona s analognog pina A1 i dodjeljivanje vrijednosti varijabli b
  delay(50);        //Stanka od 50 ms prije nastavka
  c=analogRead(A2); //Čitanje vrijednosti napona s analognog pina A2 i dodjeljivanje vrijednosti varijabli c
  delay(50);        //Stanka od 50 ms prije nastavka
  d=analogRead(A3); //Čitanje vrijednosti napona s analognog pina A3 i dodjeljivanje vrijednosti varijabli d
  delay(50);        //Stanka od 50 ms prije nastavka
  e=analogRead(A4); //Čitanje vrijednosti napona s analognog pina A4 i dodjeljivanje vrijednosti varijabli e

  Serial.print(a);  //Ispis vrijednosti varijable a
  Serial.print(" ");// Ispis praznog mjesta
  Serial.print(b); //Ispis vrijednosti varijable b
  Serial.print(" ");// Ispis praznog mjesta
  Serial.print(c); //Ispis vrijednosti varijable c
  Serial.print(" ");// Ispis praznog mjesta
  Serial.print(d); //Ispis vrijednosti varijable d
  Serial.print(" ");// Ispis praznog mjesta
  Serial.println(e); //Ispis vrijednosti varijable e i prebacivanje u novi red
  delay(50); //Stanka od 50 ms
}
```

10.1.2. Očitavanje i ispis vrijednosti u suradnji s MATLAB-om

```
//Definicija varijabli:
int mod = -1;
int vrijednost1;
int vrijednost2;
int vrijednost3;
int vrijednost4;
int vrijednost5;


void setup() {

//Započinjanje serijske komunikacije s računalom po 9600bps:
  Serial.begin(9600);

//Provjera serijske veze s računalom
  Serial.println("a"); //Ispiši u text datoteku u jednom redku samo jedan element, slovo a
  char a = 'b'; //Definiranje varijeble a formata character i dodjeljivanje vrijednosti različitog od slova a
    while (a != 'a') //Dok je god varijabla a različita od slova a čitaj vrijednost zapisanu na text datoteci,
                     //kada poprimi slovo a, izađi iz petlje
      {
        a = Serial.read();  //varijabla a poprima vrijednost iz text datoteke
      }
}

void loop() {

  if (Serial.available()>0) //Ako postoje vrijednosti za čitanje s porta
  {
    mod = Serial.read(); //Varijabla mod poprima očitanu vrijednost
    switch (mod) //Vrti petlju dok se ne ipune dolje navedeni uvjeti
    {
      case 'F': //Kada varijabla mod poprimi vrijednost F
      vrijednost1 = analogRead(A0); //Varijabla vrijednost1 poprima očitanu vrijednost s pina A0
      delay(50);                    //Čekanje 50 ms
      vrijednost2 = analogRead(A1); //Varijabla vrijednost2 poprima očitanu vrijednost s pina A1
       delay(50);                     //Čekanje 50 ms
       vrijednost3 = analogRead(A2); //Varijabla vrijednost3 poprima očitanu vrijednost s pina A2
       delay(50);                     //Čekanje 50 ms
       vrijednost4 = analogRead(A3); //Varijabla vrijednost4 poprima očitanu vrijednost s pina A3
       delay(50);                     //Čekanje 50 ms
       vrijednost5 = analogRead(A4); //Varijabla vrijednost5 poprima očitanu vrijednost s pina A4
       delay(50);                     //Čekanje 50 ms
       Serial.print(vrijednost1); //Ispiši vrijednost varijable vrijednost1 u text datoteku
       Serial.print(" ");         //Ispiši razmak
       Serial.print(vrijednost2); //Ispiši vrijednost varijable vrijednost1 u text datoteku
       Serial.print(" ");         //Ispiši razmak
       Serial.print(vrijednost3); //Ispiši vrijednost varijable vrijednost1 u text datoteku
       Serial.print(" ");         //Ispiši razmak
       Serial.print(vrijednost4); //Ispiši vrijednost varijable vrijednost1 u text datoteku
       Serial.print(" ");         //Ispiši razmak
       Serial.println(vrijednost5); //Ispiši vrijednost varijable vrijednost1 u text datoteku
                                    //i prebaci se u sljedeći red
       delay(50); //Čekaj 50 ms
       break;
     }
   }
}
```

10.2. Programski kôd u MATLAB-u

10.2.1. Program za animacijski prikaz vrijednosti senzora

```
clc;
close all;
clear all;

%Definiranje porta
comPort='COM3';

%Testiranje serijske komunikacije
if (~exist('serialFlag','var'))
    [senzor.s, serialFlag] = Uspostavljanje_veze(comPort);
end

%Postavljanje prozora i buttona
h=figure(1);

text11=uicontrol('Style', 'text', 'String', 'X: 0 stupnjeva', 'pos', [80
700 100 25], 'parent', h);
text12=uicontrol('Style', 'text', 'String', 'X: 0 stupnjeva', 'pos', [80
675 100 25], 'parent', h);
text21=uicontrol('Style', 'text', 'String', 'X: 0 stupnjeva', 'pos', [340
700 100 25], 'parent', h);
text22=uicontrol('Style', 'text', 'String', 'X: 0 stupnjeva', 'pos', [340
675 100 25], 'parent', h);
text31=uicontrol('Style', 'text', 'String', 'X: 0 stupnjeva', 'pos', [600
700 100 25], 'parent', h);
text32=uicontrol('Style', 'text', 'String', 'X: 0 stupnjeva', 'pos', [600
675 100 25], 'parent', h);
text41=uicontrol('Style', 'text', 'String', 'X: 0 stupnjeva', 'pos', [860
700 100 25], 'parent', h);
text42=uicontrol('Style', 'text', 'String', 'X: 0 stupnjeva', 'pos', [860
675 100 25], 'parent', h);
text51=uicontrol('Style', 'text', 'String', 'X: 0 stupnjeva', 'pos', [1120
700 100 25], 'parent', h);
text52=uicontrol('Style', 'text', 'String', 'X: 0 stupnjeva', 'pos', [1120
675 100 25], 'parent', h);
button=uicontrol('Style', 'togglebutton', 'String', 'Zaustavi i zatvori
serijski port', 'pos', [0 0 200 25], 'parent', h);


%Kalibracija senzora
mase=[0 250 500 750 1000]; %grama
m1=zeros(length(mase), 1)';
m2=zeros(length(mase), 1)';
m3=zeros(length(mase), 1)';
m4=zeros(length(mase), 1)';
m5=zeros(length(mase), 1)';
```

```
%Pročitaj vrijednost za svaku masu i dodjeli joj odgovarajucu silu
for i=2:length(mase)
    mbox=msgbox(['Optereti palac masom od' num2str(mase(i)) 'g']);
uiwait(mbox);
    m1(i) = Ocitaj_vrijednost_palac(senzor);
   while (m1(i) < m1(i-1)) || m1(i) == 0;
      m1(i) = Ocitaj_vrijednost_palac(senzor);
   end
end
Palac=m1;

for i=2:length(mase)
    mbox=msgbox(['Optereti kažiprst masom od' num2str(mase(i)) 'g']);
uiwait(mbox);
    m2(i) = Ocitaj_vrijednost_kaziprst(senzor);
   while (m2(i) < m2(i-1)) || m2(i) == 0;
      m2(i) = Ocitaj_vrijednost_kaziprst(senzor);
   end
end
Kaziprst=m2;

for i=2:length(mase)
    mbox=msgbox(['Optereti srednji prst masom od' num2str(mase(i)) 'g']);
uiwait(mbox);
    m3(i) = Ocitaj_vrijednost_srednji(senzor);
   while (m3(i) < m3(i-1)) || m3(i) == 0;
      m3(i) = Ocitaj_vrijednost_srednji(senzor);
   end
end
Srednji=m3;

for i=2:length(mase)
    mbox=msgbox(['Optereti prstenjak masom od' num2str(mase(i)) 'g']);
uiwait(mbox);
    m4(i) = Ocitaj_vrijednost_prstenjak(senzor);
   while (m4(i) < m4(i-1)) || m4(i) == 0;
      m4(i) = Ocitaj_vrijednost_prstenjak(senzor);
   end
end
Prstenjak=m4;

for i=2:length(mase)
    mbox=msgbox(['Optereti mali prst masom od' num2str(mase(i)) 'g']);
uiwait(mbox);
    m5(i) = Ocitaj_vrijednost_mali(senzor);
   while (m5(i) < m5(i-1)) || m5(i) == 0;
      m5(i) = Ocitaj_vrijednost_mali(senzor);
   end
end
Mali=m5;

P1=polyfit(Palac, mase, 2);
P2=polyfit(Kaziprst, mase, 2);
```

```
P3=polyfit(Srednji, mase, 2);
P4=polyfit(Prstenjak, mase, 2);
P5=polyfit(Mali, mase, 2);



%Postavljanje grafa

[xsphere, ysphere, zsphere] = sphere();

myaxes1=axes('position', [0.1 0.1 0.12 0.8],'xlim', [-20 20], 'ylim', [-20
20], 'zlim', [0 1600]);
view(3)
%grid on;
axis equal;
hold on

h(1)=surface(xsphere, ysphere, zsphere);
combinedobject1 = hgtransform('parent', myaxes1);
set(h, 'parent', combinedobject1)
drawnow


myaxes2=axes('position', [0.27 0.1 0.12 0.8],'xlim', [-20 20], 'ylim', [-20
20], 'zlim', [0 1600]);
view(3)
%grid on;
axis equal;
hold on

h(1)=surface(xsphere, ysphere, zsphere);
combinedobject2 = hgtransform('parent', myaxes2);
set(h, 'parent', combinedobject2)
drawnow


myaxes3=axes('position', [0.44 0.1 0.12 0.8],'xlim', [-20 20], 'ylim', [-20
20], 'zlim', [0 1600]);
view(3)
%grid on;
axis equal;
hold on

h(1)=surface(xsphere, ysphere, zsphere);
combinedobject3 = hgtransform('parent', myaxes3);
set(h, 'parent', combinedobject3)
drawnow


myaxes4=axes('position', [0.61 0.1 0.12 0.8],'xlim', [-20 20], 'ylim', [-20
20], 'zlim', [0 1600]);
view(3)
%grid on;
axis equal;
hold on
```

```
h(1)=surface(xsphere, ysphere, zsphere);
combinedobject4 = hgtransform('parent', myaxes4);
set(h, 'parent', combinedobject4)
drawnow


myaxes5=axes('position', [0.78 0.1 0.12 0.8],'xlim', [-20 20], 'ylim', [-20
20], 'zlim', [0 1600]);
view(3)
%grid on;
axis equal;
hold on

h(1)=surface(xsphere, ysphere, zsphere);
combinedobject5 = hgtransform('parent', myaxes5);
set(h, 'parent', combinedobject5)
drawnow




%Ocitavanje senzora, dodjeljivanje i ispis vrijednosti
while (get(button, 'value') == 0)

[masa]=Ocitaj_vrijednost(senzor);

masa1=polyval(P1, masa(5,1));
masa2=polyval(P2, masa(4,1));
masa3=polyval(P3, masa(3,1));
masa4=polyval(P4, masa(2,1));
masa5=polyval(P5, masa(1,1));

if(masa1>0)
    sila1=masa1*9.81*0.001;
else
    masa1=0.1;
    sila1=0;
end
if (masa2>0)
    sila2=masa2*9.81*0.001;
else
    masa2=0.1;
    sila2=0;
end
if (masa3>0)
    sila3=masa3*9.81*0.001;
else
    masa3=0.1;
    sila3=0;
end
if (masa4>0)
    sila4=masa4*9.81*0.001;
else
    masa4=0.1;
    sila4=0;
```

```
end
if (masa5>0)
   sila5=masa5*9.81*0.001;
else
    masa5=0.1;
    sila5=0;
end



 set(text11, 'String', ['Masa: ' num2str(round(masa1)) ' g'])
 set(text12, 'String', ['Sila: ' num2str(round(sila1)) ' N'])
 set(text21, 'String', ['Masa: ' num2str(round(masa2)) ' g'])
 set(text22, 'String', ['Sila: ' num2str(round(sila2)) ' N'])
 set(text31, 'String', ['Masa: ' num2str(round(masa3)) ' g'])
 set(text32, 'String', ['Sila: ' num2str(round(sila3)) ' N'])
 set(text41, 'String', ['Masa: ' num2str(round(masa4)) ' g'])
 set(text42, 'String', ['Sila: ' num2str(round(sila4)) ' N'])
 set(text51, 'String', ['Masa: ' num2str(round(masa5)) ' g'])
 set(text52, 'String', ['Sila: ' num2str(round(sila5)) ' N'])

 Translacija1 = makehgtform('translate', [0 0 masa1]);
        set(combinedobject1, 'matrix', Translacija1);

        scaling1 = makehgtform('scale', masa1/10);
        set(combinedobject1, 'matrix', scaling1);

        set(combinedobject1, 'matrix', Translacija1*scaling1);

Translacija2 = makehgtform('translate', [0 0 masa2]);
        set(combinedobject2, 'matrix', Translacija2);

        scaling2 = makehgtform('scale', masa2/10);
        set(combinedobject2, 'matrix', scaling2);

        set(combinedobject2, 'matrix', Translacija2*scaling2);

Translacija3 = makehgtform('translate', [0 0 masa3]);
        set(combinedobject3, 'matrix', Translacija3);

        scaling3 = makehgtform('scale', masa3/10);
        set(combinedobject3, 'matrix', scaling3);

        set(combinedobject3, 'matrix', Translacija3*scaling3);

Translacija4 = makehgtform('translate', [0 0 masa4]);
        set(combinedobject4, 'matrix', Translacija4);

        scaling4 = makehgtform('scale', masa4/10);
        set(combinedobject4, 'matrix', scaling4);

        set(combinedobject4, 'matrix', Translacija4*scaling4);

Translacija5 = makehgtform('translate', [0 0 masa5]);
        set(combinedobject5, 'matrix', Translacija5);
```

```
        scaling5 = makehgtform('scale', masa5/10);
        set(combinedobject5, 'matrix', scaling5);

        set(combinedobject5, 'matrix', Translacija5*scaling5);

 pause(0.1);
end

zatvoriPort;
```

10.2.2. Program za ispis iscrtavanje vrijednosti na grafu

```
clc;
close all;
clear all;

%Definiranje porta
comPort='COM3';

%Testiranje serijske komunikacije
if (~exist('serialFlag','var'))
    [senzor.s, serialFlag] = Uspostavljanje_veze(comPort);
end

%Postavljanje prozora i buttona
h=figure(1);

line1 = line('color', 'red');
line2 = line('color', 'blue');
line3 = line('color', 'yellow');
line4 = line('color', 'green');
line5 = line('color', 'magenta');
button=uicontrol('Style', 'togglebutton', 'String', 'Zaustavi i zatvori
serijski port', 'pos', [0 0 200 25], 'parent', h);
legend('Palac', 'Kaziprst', 'Srednji prst', 'Prstenjak', 'Mali prst')
xlabel('Vrijeme [s]')
ylabel('Sila [mN]')
i=0;

%Kalibracija senzora
mase=[0 100 200 300 400 500 600 700 800 900 1000]; %kg
m1=zeros(length(mase), 1)';
m2=zeros(length(mase), 1)';
m3=zeros(length(mase), 1)';
m4=zeros(length(mase), 1)';
m5=zeros(length(mase), 1)';

Palac = [334   347   357   363   367   367   374   377   377   383   383];
%Snimljene vrijednosti u postupku kalibracije
```

```
Kaziprst = [   335   384   390   395   395   398   408   413   425   426
428]; %Snimljene vrijednosti u postupku kalibracije
Srednji =  [376   459   480   495   519   532   533   554   557   567
567]; %Snimljene vrijednosti u postupku kalibracije
Prstenjak = [374   426   442   448   457   460   460   488   497   517
544]; %Snimljene vrijednosti u postupku kalibracije
Mali = [392   418   418   436   437   441   441   446   446   463   493];
%Snimljene vrijednosti u postupku kalibracije



P1=polyfit(Palac, mase, 3);
P2=polyfit(Kaziprst, mase, 3);
P3=polyfit(Srednji, mase, 3);
P4=polyfit(Prstenjak, mase, 3);
P5=polyfit(Mali, mase, 3);



%Ocitavanje senzora, dodjeljivanje i ispis vrijednosti
while (get(button, 'value') == 0)

[masa]=Ocitaj_vrijednost(senzor);

masa1=polyval(P1, masa(5,1));
masa2=polyval(P2, masa(4,1));
masa3=polyval(P3, masa(3,1));
masa4=polyval(P4, masa(2,1));
masa5=polyval(P5, masa(1,1));

if(masa1>0)
    sila1=masa1*9.81;
else
    masa1=0.1;
    sila1=0;
end
if (masa2>0)
    sila2=masa2*9.81;
else
    masa2=0.1;
    sila2=0;
end
if (masa3>0)
    sila3=masa3*9.81;
else
    masa3=0.1;
    sila3=0;
end
if (masa4>0)
    sila4=masa4*9.81;
else
    masa4=0.1;
    sila4=0;
end
if (masa5>0)
   sila5=masa5*9.81;
```

```
else
    masa5=0.1;
    sila5=0;
end



x1=get(line1, 'xdata');
y1=get(line1, 'ydata');
x2=get(line2, 'xdata');
y2=get(line2, 'ydata');
x3=get(line3, 'xdata');
y3=get(line3, 'ydata');
x4=get(line4, 'xdata');
y4=get(line4, 'ydata');
x5=get(line5, 'xdata');
y5=get(line5, 'ydata');

x1 = [x1 i];
y1 = [y1 sila1];
x2 = [x2 i];
y2 = [y2 sila2];
x3 = [x3 i];
y3 = [y3 sila3];
x4 = [x4 i];
y4 = [y4 sila4];
x5 = [x5 i];
y5 = [y5 sila5];


set(line1, 'xData', x1, 'yData', y1);
set(line2, 'xData', x2, 'yData', y2);
set(line3, 'xData', x3, 'yData', y3);
set(line4, 'xData', x4, 'yData', y4);
set(line5, 'xData', x5, 'yData', y5);

i=i+1;


 pause(0.001);
end

zatvoriPort;
```

10.2.3. Program za kalibriranje i aproksimaciju na izmjerenim vrijednostima

```
clc;
close all;
clear all;

%Definiranje porta
comPort='COM3';

%Testiranje serijske komunikacije
if (~exist('serialFlag','var'))
    [senzor.s, serialFlag] = Uspostavljanje_veze(comPort);
end

%Kalibracija senzora
mase=[0 100 200 300 400 500 600 700 800 900 1000]; %grama
m1=zeros(length(mase), 1)';
m2=zeros(length(mase), 1)';
m3=zeros(length(mase), 1)';
m4=zeros(length(mase), 1)';
m5=zeros(length(mase), 1)';

%Pročitaj vrijednost za svaku masu i dodjeli joj odgovarajucu silu
for i=1:length(mase)
    mbox=msgbox(['Optereti palac masom od' num2str(mase(i)) 'g']);
uiwait(mbox);
    m1(i) = Ocitaj_vrijednost_palac(senzor);
    if (i~=1)
   while (m1(i) < m1(i-1)) || m1(i) == 0;
      m1(i) = Ocitaj_vrijednost_palac(senzor);
   end
   end
end
Palac=m1;

for i=1:length(mase)
    mbox=msgbox(['Optereti kažiprst masom od' num2str(mase(i)) 'g']);
uiwait(mbox);
    m2(i) = Ocitaj_vrijednost_kaziprst(senzor);
    if (i~=1)
   while (m2(i) < m2(i-1)) || m2(i) == 0;
      m2(i) = Ocitaj_vrijednost_kaziprst(senzor);
   end
   end
end
Kaziprst=m2;

for i=1:length(mase)
   mbox=msgbox(['Optereti srednji prst masom od' num2str(mase(i)) 'g']);
uiwait(mbox);
    m3(i) = Ocitaj_vrijednost_srednji(senzor);
  if (i~=1)
   while (m3(i) < m3(i-1)) || m3(i) == 0;
      m3(i) = Ocitaj_vrijednost_srednji(senzor);
```

```
   end
  end
end
Srednji=m3;

for i=1:length(mase)
    mbox=msgbox(['Optereti prstenjak masom od' num2str(mase(i)) 'g']);
uiwait(mbox);
    m4(i) = Ocitaj_vrijednost_prstenjak(senzor);
    if (i~=1)
   while (m4(i) < m4(i-1)) || m4(i) == 0;
      m4(i) = Ocitaj_vrijednost_prstenjak(senzor);
   end
    end
end
Prstenjak=m4;

for i=1:length(mase)
    mbox=msgbox(['Optereti mali prst masom od' num2str(mase(i)) 'g']);
uiwait(mbox);
    m5(i) = Ocitaj_vrijednost_mali(senzor);
    if (i~=1)
   while (m5(i) < m5(i-1)) || m5(i) == 0;
      m5(i) = Ocitaj_vrijednost_mali(senzor);
   end
    end
end
Mali=m5;

P1=polyfit(mase, Palac, 3);
P2=polyfit(mase, Kaziprst, 3);
P3=polyfit(mase, Srednji, 3);
P4=polyfit(mase, Prstenjak, 3);
P5=polyfit(mase, Mali, 3);

masa1=polyval(P1, mase);
masa2=polyval(P2, mase);
masa3=polyval(P3, mase);
masa4=polyval(P4, mase);
masa5=polyval(P5, mase);

plot(mase, Palac, 'ro', mase, masa1);
plot(mase, Kaziprst, 'ro', mase, masa2);
plot(mase, Srednji, 'ro', mase, masa3);
plot(mase, Prstenjak, 'ro', mase, masa4);
plot(mase, Mali, 'ro', mase, masa5);

Palac
Kaziprst
Srednji
Prstenjak
Mali

if ~isempty(instrfind)
    fclose(instrfind);
    delete(instrfind);
end
```

10.2.4. Programski kôd funkcije: Uspostavljanje_veze

```
function [s, flag] = Uspostavljanje_veze(comPort)

flag = 1;
s = serial(comPort);
set(s, 'DataBits', 8);
set(s, 'StopBits', 1);
set(s, 'BaudRate', 9600);
set(s, 'Parity', 'none');

fopen(s); %Otvaranje porta
a='b'; %Varijabli a se dodjeljuje vrijednost formata %c
while (a ~= 'a') %Dok varijabla ne poprimi vrijednost "b" ulazi se u petlju
    a=fread(s, 1,'uchar');  %Čita se vrijednost varijable a iz text
datoteke, vrijednost je veličine 1 elementa i uchar preciznosti i
dodjeljuje se varijabli a
end
if (a == 'a') %Ako varijabla a poprimila vrijednost 'a'
    disp('Serijska komunikacija'); %Ispiši poruku: Serijska komunikacija
end
fprintf(s, '%c', 'a'); %U serijski port upisuje se slovo "a" varijable a,
formata c.
mbox=msgbox('Serijska komunikacija je uspostavljena.'); uiwait(mbox);
%Ispiši poruku da je uspostavljena serijska komunikacija i čekaj dok se ne
zatvori prozor
fscanf(s, '%u'); %Pročitaj zapisanu vrijednost u serijskom portu i pretvori
ju u decimalni zapis
end
```

10.2.5. Programski kôd funkcije: zatvoriPort

```
clear all
if ~isempty(instrfind)
    fclose(instrfind);
    delete(instrfind);
end
close all
disp('Serijski port zatvoren')
```

10.2.6. Programski kôd funkcije: Ocitaj_vrijednost_palac

```
function [vrijednost] =Ocitaj_vrijednost_palac(analogni_pin)
fprintf(analogni_pin.s, 'F'); %Upiši u text datoteku slovo F
vrijednost0=fscanf(analogni_pin.s, '%d'); %Očitaj vrijednost iz text
datoteke u decimalnom zapisu
vrijednost=vrijednost0(5,1);
end
```

10.2.1. Programski kôd funkcije: Ocitaj_vrijednost_kaziprst

```
function [vrijednost] =Ocitaj_vrijednost_kaziprst(analogni_pin)
fprintf(analogni_pin.s, 'F'); %Upiši u text datoteku slovo F
vrijednost0=fscanf(analogni_pin.s, '%d'); %Očitaj vrijednost iz text
datoteke u decimalnom zapisu
vrijednost=vrijednost0(4,1);
end
```

10.2.2. Programski kôd funkcije: Ocitaj_vrijednost_rednji

```
function [vrijednost] =Ocitaj_vrijednost_srednji(analogni_pin)
fprintf(analogni_pin.s, 'F'); %Upiši u text datoteku slovo F
vrijednost0=fscanf(analogni_pin.s, '%d'); %Očitaj vrijednost iz text
datoteke u decimalnom zapisu
vrijednost=vrijednost0(3,1);
end
```

10.2.3. Programski kôd funkcije: Ocitaj_vrijednost_prstenjak

```
function [vrijednost] =Ocitaj_vrijednost_prstenjak(analogni_pin)
fprintf(analogni_pin.s, 'F'); %Upiši u text datoteku slovo F
vrijednost0=fscanf(analogni_pin.s, '%d'); %Očitaj vrijednost iz text
datoteke u decimalnom zapisu
vrijednost=vrijednost0(2,1);
end
```

10.2.4. Programski kôd funkcije: Ocitaj_vrijednost_mali

```
function [vrijednost] =Ocitaj_vrijednost_mali(analogni_pin)
fprintf(analogni_pin.s, 'F'); %Upiši u text datoteku slovo F
vrijednost0=fscanf(analogni_pin.s, '%d'); %Očitaj vrijednost iz text
datoteke u decimalnom zapisu
vrijednost=vrijednost0(1,1);
end
```