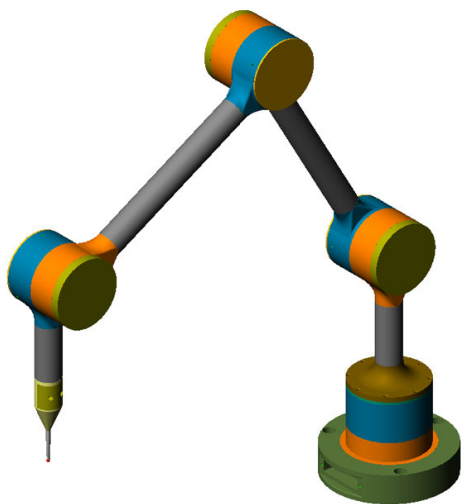




## 2 Mehanska struktura

Mehanizem roke mora omogočati čim lažje in natančnejše pozicioniranje tipala na vrhu. Za doseganje poljubne orientacije krogelnega tipala sicer potrebujemo pet prostostnih stopenj, vendar smo se zaradi enostavnejše izvedbe odločili za prototip s štirimi rotacijskimi sklepi, kar zadošča za večino meritev. Ker je mehanizem popolnoma pasiven, je pomembna majhna teža, hkrati pa moramo zagotoviti čim večjo rigidnost konstrukcije. Za izdelavo segmentov smo tako izbrali kompozit iz karbonskih vlaken, ki ima najugodnejše razmerje med težo in elastičnim modulom, sklepi pa so izdelani iz aluminija. Jeklen je le podstavek baze, saj mora omogočati trdno pričvrstitev roke na merilno površino (z vijaki ali magnetno). Dolžina večjih segmentov je 325 mm, kar pomeni da ima roka sferičen delovni prostor s premerom 1300 mm. CAD model roke je prikazan na sliki 2.



Slika 2: CAD model roke

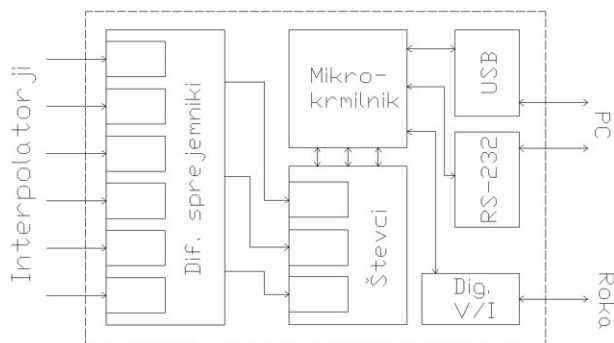
Kovinsko tipalo s sferično konico je nepremično in opremljeno s tipko za sproženje meritve in signalno LED. Na ta način je možno drsenje po merjenem objektu oz. skeniranje. Kot druga možnost se ponuja uporaba na dotik občutljivega tipala, ki avtomatsko sproži enkratni zajem pozicije, zagotavlja večjo točnost in ponovljivost.

V vsakem od sklepov je vgrajen standarden, komercialno dosegljiv optični enkoder. Trenutno so v uporabi dajalniki s 3600

črticami na obrat in tokovnimi sinusnimi izhodnimi signali. Ti se preko konektorja na bazi roke vodijo v elektronsko enoto z interpolatorji, ki resolucijo povečajo za faktor 25, kar ob upoštevanju kvadrature načina štetja daje končno resolucijo  $0.001^\circ$  za vsak sklep. To v najslabšem primeru pri popolnoma iztegnjena roki pomeni 3D resolucijo pozicije tipala  $\delta P=0.017$  mm.

## 3 Vmesniška plošča

Naloga elektronskega vmesnika je obdelava signalov iz enkoderjev ter posredovanje informacij osebemu računalniku, v katerem teče uporabniški program. Izdelan je na plošči euro formata in je zamišljen kot univerzalni vmesnik med največ šestimi enkoderji ter poljubnim osebnim računalnikom. Poleg tega vsebuje še vezje za sprejem in oddajo nekaj pomožnih digitalnih signalov za potrebe tipala, signalnih LED diod itd. Z računalnikom lahko komunicira preko RS-232 ali USB vodila. Bločni diagram vmesnika je prikazan na sliki 3.



Slika 3: Bločni diagram vmesnika

Diferencialni sprejemniki na vhodu vršijo pretvorbo signalov iz posameznih enkoderjev, ki prispejo po paricah: fazi A in B, indeksni signal ter signal napake, ki ga generira interpolator (v primeru prekinitve povezave, napake na svetilki, umazanije na enkoderskem disku ipd.). V primeru omenjenih motenj se štetje impulzov prekine in preko osebnega računalnika obvesti uporabnika.

Impulzi se nato vodijo v dekodirnike pozicije, ki so izvedeni z integriranimi vezji

LS7266R1. To so 24 bitni števcji z dodatno logiko za različne načine štetja (v našem primeru kvadraturno), resetiranje ob indeksnem pulzu, zajem stanja števcjev v določenem trenutku, upoštevanja modula štetja in drugo. Vhodni signali faz A in B se filtrirajo s pomočjo ure s frekvenco 16 Mhz. Pulzi s prekratko periodo, ki jih generira interpolator in so lahko posledica motenj v merilnem signalu, na ta način generirajo napako, o kateri je prav tako obveščen uporabnik.

Jedro vezja predstavlja mikrokrmilnik AVR proizvajalca Atmel. Ta skrbi za izmenjavo podatkov z osebnim računalnikom, inicializacijo in komunikacijo z dekoderji pozicije, nadzoruje stanje indeksnih signalov, vhodno-izhodnih linij za tipke in signalne LED itd. Prav tako nadzoruje vir in način proženja meritve ter zagotavlja hkraten zajem pozicije vseh sklepov.

UART vmesnik za RS-232 prenos je vgrajen že v mikrokrmilnik in potrebna je le še pretvorba napetostnih nivojev. Za USB povezavo pa smo uporabili modul z FTDI čipom USB245BM. Ta prevzame vse funkcije povezane z USB protokolom in omogoča hitrosti prenosa do 1 MB/s.

#### 4 Uporabniška programska oprema

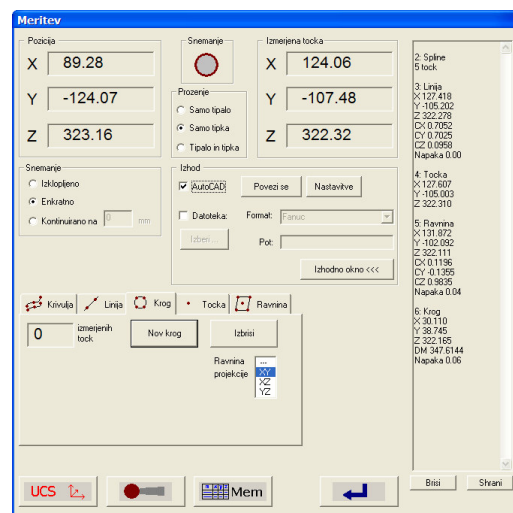
Razvili smo tudi aplikacijo, ki omogoča upravljanje z roko, zajem in obdelavo meritev. Izvaja se lahko v poljubnem osebnem računalniku z operacijskim sistemom Windows 98 / 2000 / XP. Napisana je programskem jeziku Visual C++. Program je zamišljen tako, da omogoča le osnovne funkcije, ki so potrebne, da na primeren način izmerimo določen objekt. Izmerjeni podatki se nato prenesejo v katerega od obstoječih paketov za CAD.

Tako ostane vmesnik za merjenje enostaven za uporabo in se nanj hitro privadimo, saj je vsa manipulacija nad izmerjenimi geometrijskimi primitivi prepuščena CAD paketu, ki ga običajno uporabljamo.

Primer zaslona med merjenjem je na sliki 4. Trenutno je mogoče poleg merjenja točk, prostih krivulj in površin le merjenje nekaj osnovnih objektov – ravnin, ravnih linij, krogov. Vse lahko izmerimo s poljubnim številom točk,

s čimer lahko povečamo natančnost meritve in ugotovimo geometrijsko odstopanje merjene oblike od idealne. Omogoča nam tudi poljubno postavitev koordinatnega sistema glede na izmerjene objekte in nastavitve vira in načina proženja. Ta sta lahko dva:

- Enkratno proženje – posamezna pozicija tipala se zajame v trenutku pritiska tipke oz. dotika tipala.
- Kontinuirano proženje – to je namenjeno zajemanju večjega števila točk. Dokler držimo tipko, se točke neprekinjeno snemajo z gostoto, ki jo določimo preko najmanjše razdalje med njimi. Na ta način lahko digitaliziramo do 300 točk na sekundo.



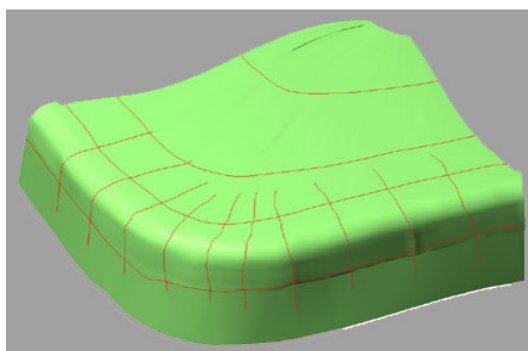
Slika 4: Zaslona med merjenjem

Vse meritve se izpisujejo na zaslonu in/ali zapisujejo v datoteko. Grafični prikaz in nadaljna obdelava pa sta prepuščena obstoječim CAD paketom; trenutno je izvedena povezava s paketom AutoCAD. Z njim se naša aplikacija povezuje preko ActiveX vmesnika. Njegove temeljne funkcije so:

- Vzpostavitev povezave z novo ali že obstoječo sejo AutoCADa. V zadnjem primeru se izmerjeni objekti vstavljajo v trenutno odprti dokument, kar enostavno omogoča hitro primerjavo merjenca in njegovega CAD modela.
- Risanje vseh izmerjenih objektov. To vključuje izračun in pretvorbo potrebnih podatkov ter kreiranje objektov v AutoCADu po končani meritvi. V primeru merjenja krivulj in površin je možen prikaz napredka

med samim merjenjem, saj se le-te obnavljajo v realnem času.

Primer meritve je prikazan na sliki 5, kjer na eni strani vidimo izdelek in na drugi strani njegovo rekonstrukcijo v AutoCADu - digitalizirane krivulje skupaj z avtomatsko generirano površino.



Slika 5: Merjeni izdelek in del rekonstruirane površine

## 5 Kalibracija

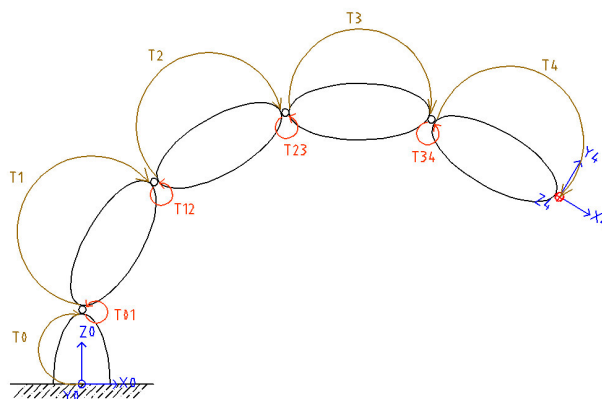
Pri izdelavi posameznih delov mehanizma neizogibno pride do odstopanj nekaterih dimenzij od predpisanih. Tudi najmanjše napake v sklepih imajo velik končen vpliv pri kinematičnem izračunu pozicije vrha. Očitno je torej, da je potrebno teoretični model roke optimirati oz. uskladiti njegove parametre z dejansko geometrijo mehanizma.

Prvi korak v postopku kalibracije je izbira primernega modela, ki opisuje kinematiko mehanizma. Ta naj bi imel predvsem dve pomembni lastnosti:

- Popolnost – opisati mora vsako možno variacijo k kinematiki fizičnega mehanizma.
- Proporcionalnost – majhne spremembe v fizični geometriji naj povzročijo tudi majhne spremembe v parametrih modela. Modeli, ki te lastnosti nimajo, so težavni za optimizacijo, saj povzročajo nestabilnost optimizacijskih algoritmov [2].

Klasičen Denavit-Hartenbergov model, ki ga najpogosteje uporabljamo za zapis podobnih mehanizmov, teh lastnosti nima, saj predpisuje le štiri parametre za vsak segment, poleg tega pa je nezvezen za konfiguracije s paralelnimi ali skoraj paralelnimi sklepi. S tesnejšo povezanostjo parametrov modela in fizičnih parametrov se temu izognemo.

Zato smo za opis roke uporabili model, kjer vsakemu segmentu predpišemo dva koordinatna sistema (slika 6). Transformacijo med njima opisujejo homogene matrice  $T_i$ , ki so postavljene v osi sklepov, transformacijo  $T_{i-1,i}$  med enim in drugim segmentom pa predstavlja le rotacija okrog Z osi.



Slika 6: Kalibracijski model roke

Vsak segment je torej opisan s šestimi parametri, ki predstavljajo tri translacije in tri zaporedne rotacije - Eulerjeve ZYX kote [3]. Lega tipala glede na osnovni koordinatni sistem je tako podana z:

$$T_{04} = T_0 \cdot T_{01} \cdot T_1 \cdot T_{12} \cdot T_2 \cdot T_{23} \cdot T_3 \cdot T_{34} \cdot T_4 \quad (1)$$

Parametri modela in s tem elementi matrik  $T_0$ ,  $T_1$ ,  $T_2$ ,  $T_3$ ,  $T_4$  se optimirajo glede na podatke pridobljene s kalibracijo in nato ostanejo nespremenjeni. Pri vsakokratnem izračunu pozicije tipala pa se glede na kote v sklepih

izračunajo le vrednosti matrik  $T_{01}$ ,  $T_{12}$ ,  $T_{23}$ ,  $T_{34}$ .

Postopek same kalibracije zahteva hkratno merjenje pozicije vrha tipala in vrednosti kotov v sklepih. Da bi se izognili zahtevnemu merjenju natančne 3D pozicije tipala v velikem številu točk [4], smo izdelali ploščo s koničnimi izvrtinami v znanih pozicijah, v katere lahko sferično tipalo postavimo v različnih orientacijah. Nad tako dobljeno množico točk nato izvedemo identifikacijski algoritem, ki naj čim bolj zmanjša razliko med dejansko pozicijo tipala in vrednostmi, izračunanimi z uporabljenim modelom. Kot merilo natančnosti modela oz. cenilno funkcijo uporabimo vsoto kvadratov preostankov, t.j. razlik med dejanskimi in izračunanimi pozicijami v kalibracijskih točkah:

$$\Phi = \sum_{i=1}^n (x'_i - x)^2 + (y'_i - y)^2 + (z'_i - z)^2 \quad (2)$$

Z iterativnim spreminjanjem parametrov in izračunavanjem cenilke nato skušamo le-to zmanjšati. Ker so koordinate točk in posledično cenilka nelinearne funkcije parametrov modela in kotov v sklepih, bodo izbira modela, začetne vrednosti parametrov in algoritem kalibracije vsi vplivali na stabilnost in končno uspešnost postopka.

Preiskusili smo več algoritmov za optimizacijo parametrov, ki v osnovi temeljijo na gradientni metodi iskanja minimuma cenilke [5]. Za najbolj uporabnega se je izkazal Levenberg – Marquardtov postopek, ki v vsakem iteracijskem ciklu vektor parametrov modela  $p$  popravi za vrednost

$$\Delta p = -(H + \lambda I)^{-1} \cdot g \quad , \quad (3)$$

kjer je  $g$  vektor gradienta cenilne funkcije,  $H$  pa je matrika drugih odvodov in vsebuje informacijo o njeni ukrivljenosti.  $\lambda$  je spremenljiv skalar, ki vpliva na stabilnost algoritma.

Seveda se lahko tudi s kalibracijo le bolj ali manj približamo resničnemu odzivu mehanizma. Njegov model je namreč nepopoln v smislu nezmožnosti opisa negeometrijskih napak – elastičnosti segmentov, zračnosti in

ekscentričnosti v sklepih, nelinearnosti merjenja kota in drugih. Tako preostanek po enačbi (2) ne konvergira k 0, temveč k neki končni vrednosti, ki predstavlja neujemanje modela v kalibracijskih točkah.

## 6 Zaključek

Naprava ima kot prototip pri uporabi še nekatere omejitve, ki izvirajo predvsem iz slabše ponovljivosti mehanizma, saj ta ni bil grajen za doseganje visoke točnosti. V tem pogledu sledijo še številne izboljšave – od preciznejše mehanike do upoštevanja in obvladovanja večine zunanjih in notranjih motilnih vplivov – ki ob nekaterih dopolnitvah elektronske in programske opreme obetajo nastanek resnično uporabnega inštrumenta.

## Literatura

- [1] I. Kovač, Antropoidne merilne naprave v koordinatni merilni tehniki, *Strojniški vestnik* (44), str. 175-182, 1998
- [2] L. Sciavicco, B. Siciliano, *Modelling and Control of Robot Manipulators*, Springer-Verlag, London, 2000
- [3] A. Goswami, A. Quaid, M. A. Peshkin, Complete parameter identification of a robot using partial pose information, *IEEE Control Systems*, Vol. 13, No. 5, str. 6-14, October 1993
- [4] R. Bernhardt, *Robot Calibration*, Chapman & Hall, London, 1993
- [5] R.J. Horning, A comparison of identification techniques for robot calibration, M.S. Thesis, Case Western Reserve University, 1998