



Zoran Dimić, Miloš Glavonjić, Dragan Milutinović, Saša Živanović, Vladimir Kvirgić, Marija Milićević¹⁾

UPRAVLJAČKI SISTEM OTVORENE ARHITEKTURE ZA UPRAVLJANJE TROOSNE MAŠINE SA PARALELNO KINEMATIKOM²⁾

Rezime

U radu je prikazano konfigurisanje sistema otvorene arhitekture za upravljanje troosnom mašinom sa paralelnom kinematikom. Na realnoj mašini je testiran softver i hardver, koji može da se koristi i za višeosno upravljanje mašina alatki i industrijskih robota. Ostvareno je upravljanje servo motorima na mašini, na PC platformi pod Linux operativnim sistemom.

Ključne reči: upravljanje, mašina sa paralelnom kinematikom, servo motori, otvorena arhitektura.

1. UVOD

Upravljanje otvorene arhitekture (*Open architecture control - OAC*) je dobro poznat pojam u oblasti upravljanja mašinama alatkama. Na sličan način na koji je otvorena arhitektura uticala na revolucionaran rast PC industrije, otvorenost može revolucionarno unaprediti CNC industriju.

Iako otvorena arhitektura (otvoren sistem) nije novi koncept u oblasti softverskog inženjerstva, još uvek nije ustanovljena jasna definicija upravljanja otvorene arhitekture. Mnoge istraživačke grupe, kao što su *OMAC (Open Modular Architecture Controllers)* u SAD-u, *OSAKA (Open System Architecture for Controls within Automation System)* u E.U. i *JOP (Japanese Open Promotion Group)* u Japanu, imaju svoj stav prema pomenutim sistemima.

Softverski sistemi otvorene arhitekture postaju značajan deo proizvodnih sistema u kojima korisnici, integratori sistema i softverski inženjeri koji učestvuju u razvoju istih, direktno mogu uticati na konfiguraciju sistema, u trenutku kada tu fleksibilnost zahteva sam proizvodni proces.

Kada je otvorenost arhitekture u pitanju, upravljački sistemi se mogu svrstati u sledeće tri kategorije:

1. sistemi sa otvorenim korisničkim interfejsom;
2. sistemi sa ograničenom otvorenosću upravljačkog jezgra;
3. sistemi sa potpuno otvorenom arhitekturom.

Hardver upravljačkog sistema zasnovan na PC računarskoj platformi, odgovarajuće interfejs kartice za kontrolu kretanja i softverski orijentisani CNC su tri glavne komponente upravljačkih sistema otvorene arhitekture koji su danas prisutni na domaćem i svetskom tržištu. Kod komercijalno raspoloživih CNC sistema, kao što su sistemi iz *FANUC*-ove serije 150/160/180/210, računarski sistem, iako je baziran na PC arhitekturi, predstavlja za korisnike "crnu kutiju". U ovom slučaju, PC računarska platforma, koja je sastavni deo upravljačke jedinice, postavljena je sa zadatkom da poboljša i učini fleksibilnim interfejs prema korisniku. I pored toga, takav CNC se ne može svrstati u grupu sistema otvorene arhitekture u strogom smislu. Komercijalno raspoloživi CNC sistemi se sastoje od:

1. standardne PC platforme;
2. odgovarajućeg hardvera za numeričku kontrolu kretanja i
3. kartice, obično na bazi nekog DSP-a (procesora za obradu signala u realnom vremenu), koja treba da obezbedi strogi determinizam u izvršavanju vremenski kritičnih procesa.

¹⁾ Zoran Dimić, dipl. inž. el. (zoran.dimic@li.rs), dr Vladimir Kvirgić, LOLA institut, (vladimir.kvirgic@li.rs), Marija Milićević, dipl. inž. el., (marija.milicevic@li.rs), LOLA institut, Beograd, prof. dr Miloš Glavonjić, (mglavonjic@mas.bg.ac.rs), prof. dr Dragan Milutinović, (dmilutinovic@mas.bg.ac.rs), doc. dr Saša Živanović, (szivanovic@mas.bg.ac.rs), Univerzitet u Beogradu, Mašinski fakultet.

²⁾ Rađeno u okviru projekta TR14202 "Razvoj upravljačke jedinice otvorene arhitekture za upravljanje mašinama alatkama i industrijskim robotima", koji je finansiralo Ministarstvo za nauku i tehnološki razvoj Republike Srbije

Na *PC* platformi (procesoru) se izvršavaju procesi koji nisu vremenski kritični. Dva procesora, *PC* i *DSP*, komuniciraju međusobno ili posredstvom *PC* magistrale ili preko dvoportnog *RAM*-a. Navedena konfiguracija obezbeđuje poboljšanje korisničkog interfejsa, veću fleksibilnost sistema, kako za proizvođače mašina alatki, tako i za krajnje korisnike.

Softverske metode za implementaciju upravljačkih sistema otvorene arhitekture su različite koliko i definicije ovih sistema. Trenutno dostupni *OAC* sistemi na tržištu, kao i razvoj *OAC* sistema, uglavnom su bazirani na *Microsoft Windows* tehnologijama ili *Microsoft Windows real-time* tehnologijama:

1. *MDSI Open CNC*;
2. *Fanuc 210i/210is*;
3. *Allen Bradley 9/PC*;
4. *Siemens E&A 840D/840Di* . . .

Iako rad na razvoju *CNC* sistema pod *Linux*-om nije zastupljen koliko ekvivalentna istraživanja na *Windows* osnovi, razvoj softverski orijentisanog *CNC* otvorene arhitekture na *Linux/RTLinux* osnovi treba da donese višestruke prednosti u odnosu na ekvivalentne *Windows* sisteme. Između zajedničkih zahteva koji se danas postavljaju pred softverske sisteme otvorene arhitekture, a koji treba da karakterišu i softverski orijentisani *CNC* na *Linux*-u, ističu se:

1. Robusna, pouzdana arhitektura;
2. Upravljanje otkazom, ograničenje prostiranja greške na jedan proces;
3. Dostupnost komunikacionih protokola, alata i drajvera;
4. Podrška za izabranu procesorsku arhitekturu;
5. Različitost opcija i pripadajućih alata za izabranu distribuciju *real-time Linux* operativnog sistema;
6. Standardan *API* (*Application Programming Interface*) koji omogućava jednostavan transfer razvijenog softvera na *embedded* sisteme.

2. SOFTVERSKI ALATI ZA REALIZACIJU UPRAVLJAČKOG SISTEMA

Windows i *Linux* su dva najzastupljenija operativna sistema koja se izvršavaju na *PC* računarima. Da bi mogli da se koriste u *real-time* aplikacijama, pomenuti operativni sistemi moraju biti softverski nadograđeni tzv. *real-time* ekstenzijama. *VenturCom*-ov *RTX*, *RadisyS*-ov *Intime* i *Imagination System*-ov *Hyperkernel* predstavljaju neke od *real-time* ekstenzija za *Windows*.

U slučaju *Linux*-a, dva su pristupa trenutno aktuelna koja dovode do *real-time* sistema:

1. unapređenje mehanizma za dodelu resursa procesu i
2. apstrahovanje prekida.

Prvi pristup je implementiran u firmi *TimeSys*. *TimeSys Linux* predstavlja standardni *Linux* sa nadgrađenim kernelom, u smislu unapređenog *preemptive* algoritma za raspoređivanje proseca, koji obezbeđuje vremenski determinizam prilikom izvršavanja softverskih aplikacija. *RTLinux* koristi drugi pristup, kod koga je virtuelni hardverski nivo postavljen između standardnog *Linux* kernela i hardvera računara. Na ovaj način, implementacija *real-time* sistema postaje nezavisna od standardnog *Linux*-a. Ovo *RTLinux* čini idealnim okruženjem u kome je izvršavanje standardnog *Linux*-a podređeno malom *real-time*, strogo vremenski determinisanom sistemu.

RTLinux je *real-time* ekstenzija otvorene arhitekture koja je široko korišćena u razvoju komercijalnih, industrijskih, akademskih i naučnih aplikacija. Najduži mogući odziv *RTLinux*-a, od trenutka prepoznavanja hardverskog prekida od strane procesora, do momenta kada se pokrene prekidna rutina, iznosi manje od 10 mikrosekundi na *x86* procesorskoj arhitekturi. Na istoj arhitekturi procesi se sa reda čekanja pokrenu za manje od 25 mikrosekundi od trenutka kada ih raspoređivač procesa aktivira.

Kod softverski orijentisanih *CNC* sistema, operativni sistem ima jednu od najznačajnijih uloga. *Linux* sa *RTLinux*-om, kao *real-time* operativni sistem, u najvećoj meri zadovoljava postavljene zahteve. Na prvom mestu, izvanredne *real-time* performanse *RTLinux*-a, u potpunosti ispunjavaju zahteve koje postavlja softverski *CNC*. Pored toga, *Linux* nudi mogućnost pristupa mreži, pristup grafičkim elementima korisničkog interfejsa, upravljanju datotekama i bazama podataka, kao i drugim servisima koji nisu jednostavni za implementaciju.

Linux, po svojoj prirodi, nudi bogato programsko okruženje. Nekoliko viših programskih jezika je dostupno u *Linux* okruženju. Od ponuđenih, programski jezici kompajlerskog tipa, *C* i *C++*, nude odlične performanse i imaju pun pristup operativnom sistemu, ali nisu pogodni za razvoj grafičkih interfejsa. *RTLinux* je najvećim delom realizovan u *C*-u. *Tcl/TK* i *Python* su programski jezici interpreterskog tipa, koji su izuzetno pogodni za razvoj grafičkih interfejsa, ali imaju ograničen pristup operativnom sistemu i

hardverskim resursima. Pored pomenutih razvojnih alata, u realizaciji projekta je korišćen i *OpenGL* za virtuelnu simulaciju mašina alatki i robota.

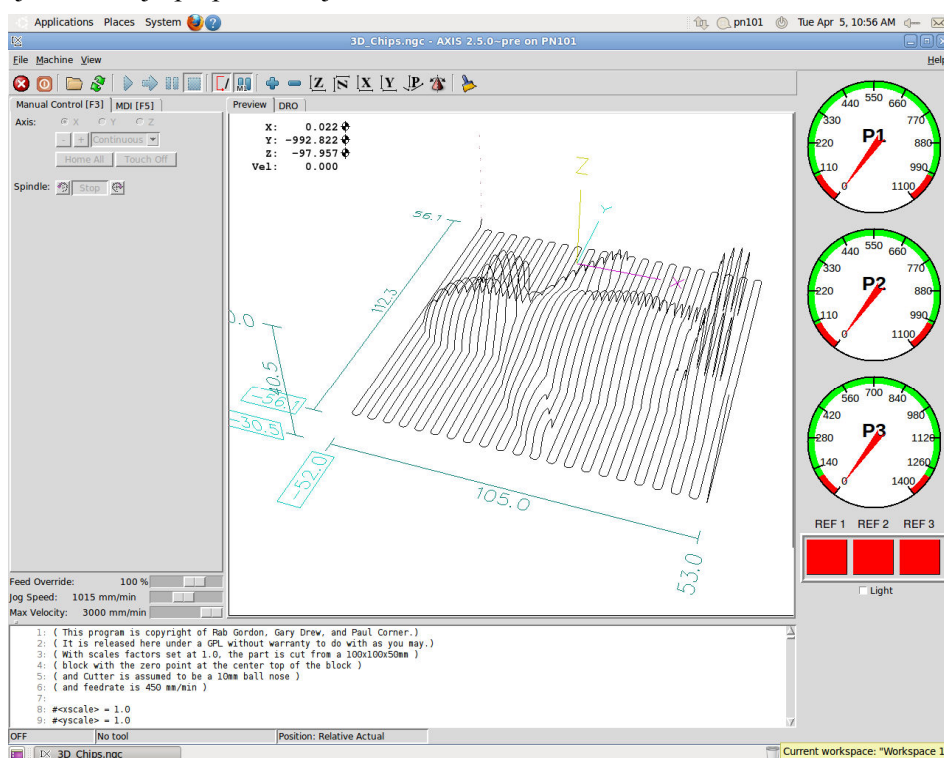
3. RAZVOJ PROGRAMSKOG SISTEMA UPRAVLJAČKE JEDINICE

Za razvoj programskog sistema upravljačke jedinice, kao osnova je poslužio softverski sistem otvorene arhitekture *EMC2 (Enhanced Machine Controller)*[1-3]. Nadgradnja sistema je sadržana u sledećem:

- Prilagodjenje korisničkog interfejsa;
- Definisanje funkcija inverzne i direktne kinematike;
- Izrada konfiguracionih datoteka za identifikaciju mašine;
- Definisanje veza između hardverskog interfejsa za kontrolu kretanja i softverskog modula *EMC*-a za upravljanje kretanjem;

3.1. Prilagodjenje korisničkog interfejsa

EMC sistemski softver ima mogućnost korišćenja različitih grafičkih interfejsa, kao što su: *Axis*, *Mini*, *TkEmc* i drugi. Najviše je korišćeno *Axis* okruženje, koje je vrlo intuitivno, sa prepoznatljivim grafičkim elementima, koji olakšavaju prepoznavanje komandi, slika 1.



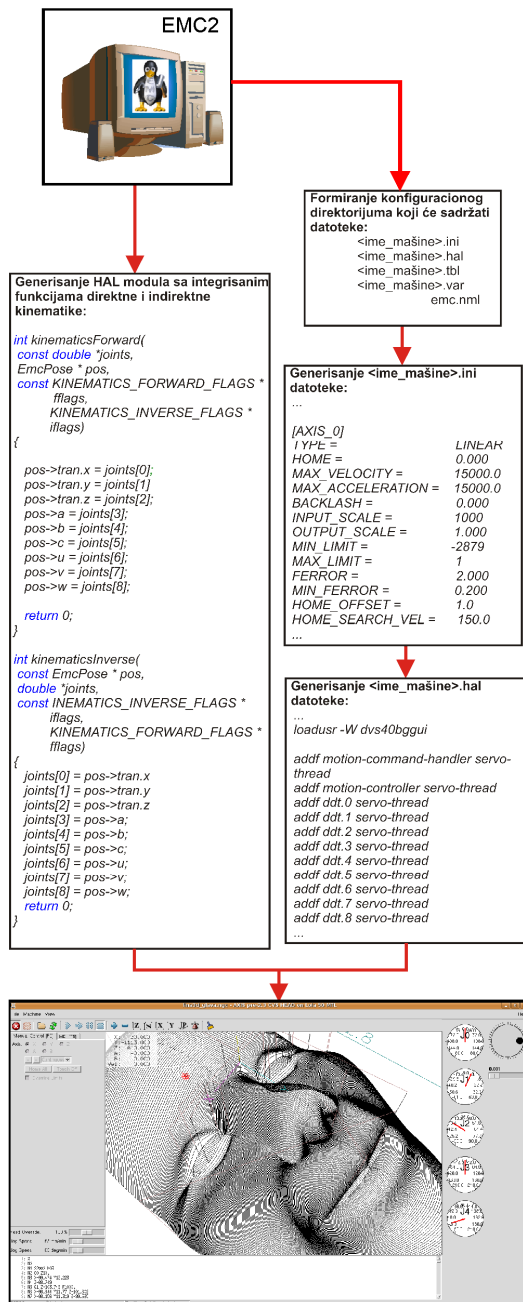
Slika 1. *AXIS* grafički interfejs sa dodatim elementima za praćenje rada pogonskih osa p1, p2 i p3 na mašini LOLA pn101_4 V.2

AXIS interfejs je za mašinu sa paralelnom kinematikom prilagođen dodavanjem vizuelnih pokazivača unutrašnjih koordinata svih pogonskih osa u milimetrima, u zadatim granicama. Na taj način se vizuelno kontroliše položaj svake ose u realnom vremenu i uočava kada je neka osa došla do granične pozicije. Prilikom simulacije programa obrade, dolazi do zaustavljanja daljeg izvršenja ukoliko bilo koja od osa prekorači svoje granice. Na taj način se obezbeđuje da se na mašini neće izvršiti program koji je može dovesti u kolizionu situaciju. Interfejs sadrži i elemente za signalizaciju ostvarenog referentnog položaja, koji su od značaja prilikom odlaska u referentni položaj mašine alatke sa specifičnom kinematikom, kakva je i razmatrana troosna mašina sa paralelnom kinematikom. Interfejs *AXIS*, kao i pomenuti specijalno definisani elementi, razvijeni su u *Python* objektno orijentisanom programskom jeziku.

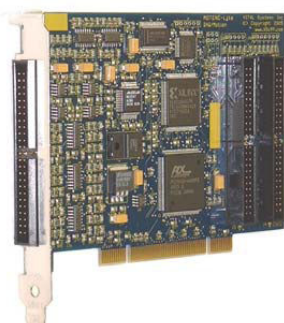
3.2. Definisanje funkcija inverzne i direktne kinematike

Zamišljen kao univerzalni interfejs prema hardveru računara, *HAL (Hardware Abstraction Layer)*. Omogućava da se konfigurisanje i nadgradnja softvera u najvećoj meri obavlja nezavisno od toga na kojoj će

se računarskoj konfiguraciji realizovani CNC softver izvršavati. *HAL* komponente predstavljaju programski okvir za integraciju funkcija inverzne i direktne kinematike u sistem za upravljanje.



Slika 2. Konfiguracioni fajlovi upravljačkog sistema



Slika 3. Interfejs kartica za kontrolu kretanja

Za upravljanje mašinama sa netrivialnom kinematikom, neophodno je izgraditi komponentu koja sadrži funkcije inverzne i direktne kinematike i integrisati je u *HAL*. Da bi se ovo učinilo, potrebno je napisati odgovarajuće funkcije u *C* programskom jeziku, prevesti i povezati datoteke sa odgovarajućim *HAL*-ovim komponentama. Sa pripadajućim **.ini* i **.hal* datotekama ovo čini jednu softversku konfiguraciju mašine, slika 2. U jezgro EMC2 upravljačkog softvera troosne mašine sa paralelnom kinematikom su integrisana rešenja inverznog i direktne kinematičkog problema [4].

3.3. Izrada konfiguracionih datoteka za identifikaciju mašine

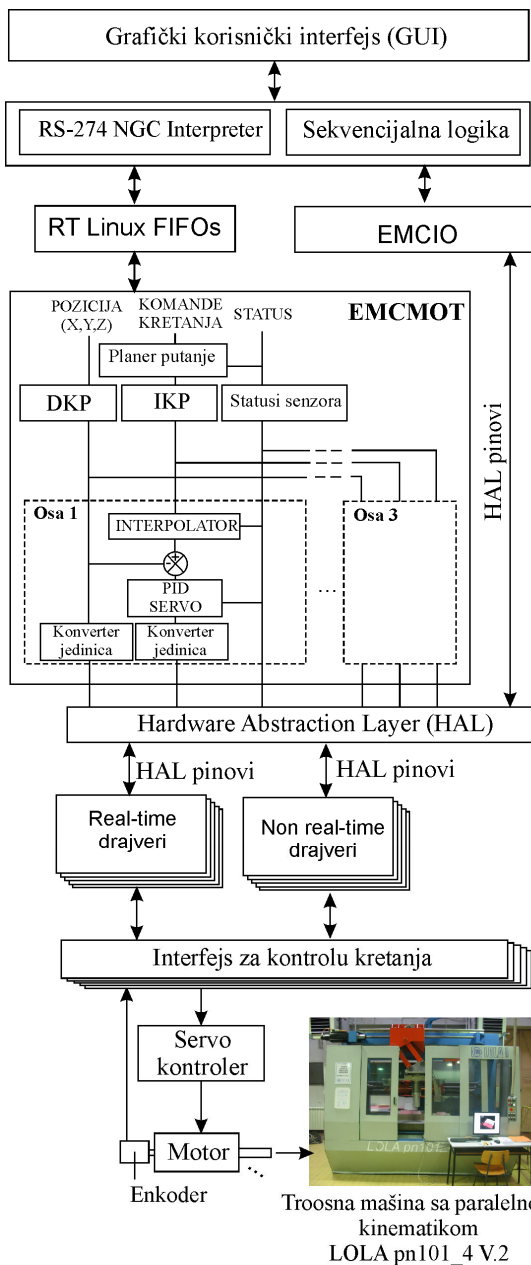
Sve ono što nije vezano za inverznu i direktnu kinematiku, a služi da se u tehničkom smislu opiše mašina, sadržano je u konfiguracionim datotekama, slika 2. Datoteka **.ini* sadrži sledeće:

1. Konfiguracione parametre *RS274NGC* interpretera;
2. Parametre koji definišu rad sekvencijalne logike;
3. Parametre vezane za performanse sistema, servo petlje i planera putanje;
4. Parametre koji se uzimaju u obzir prilikom planiranja putanje, a koji uključuju broj kontrolisanih osa u sistemu, njihove radne i granične opsege kretanja, brzina i ubrzanja itd;
5. Definiciju načina inicijalizacije mašine;
6. Parametre servo sistema, *PID* pozicione servo petlje, za svaku osu posebno, itd.

3.4. Definisane veza između hardverskog interfejsa za kontrolu kretanja i softverskog modula EMC-a za upravljanje kretanjem

U cilju definisanja veza između odabranog *MOTENC Lite* hardverskog interfejsa za kontrolu kretanja, slika 3, i *EMCMOT* softverskog modula za upravljanje kretanjem, slika 4, neophodno je formirati sledeće dve datoteke:

1. *motenc_motion.hal*, glavnu konfiguracionu datoteku koja podiže drajver za *MOTENC Lite* karticu za i postavlja veze koje omogućavaju upravljanje kretanjima troosne mašine sa paralelnom kinematikom. Deo ove datoteke koji postavlja veze ka *D/A* konvertorima i enkoderima prikazan je na slici 5.
2. *motenc_io.hal*, datoteku koja definiše ulazno-izlazne veze, kao što su veze ka krajnjim graničnim prekidačima ili referentnim sensorima troosne mašine sa paralelnom kinematikom. Na slici 6, prikazan je deo ove datoteke koji definiše vezu jedne ose sa krajnjim graničnim prekidačima i inicijalizacionim sensorima.



Slika 4. Softverski orijentisani CNC

```
# Connect PID output signals to DACs.
linksp Joint0 => motenc.0.dac-00-value
linksp Joint1 => motenc.0.dac-01-value
linksp Joint2 => motenc.0.dac-02-value

# Connect position feedback signals to encoders.
linksp Joint0-fb <= motenc.0.enc-00-position
linksp Joint1-fb <= motenc.0.enc-01-position
linksp Joint2-fb <= motenc.0.enc-02-position
```

Slika 5. Definisane veze ka D/A konvertorima i enkoderima

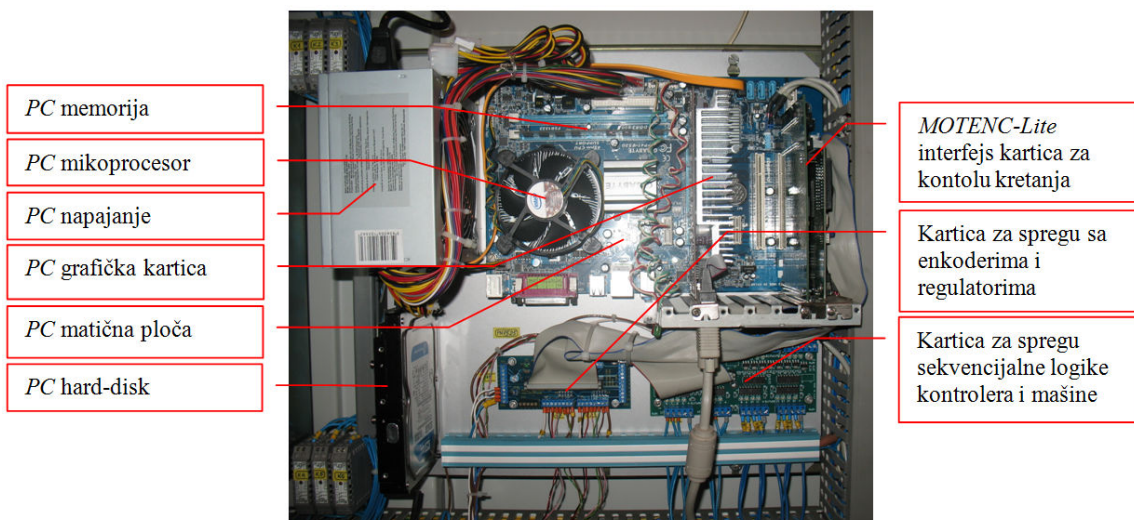
```
# Connect limit/home switch outputs to motion controller.
newsig Xminlim bit
newsig Xmaxlim bit
newsig Xhome bit
linksp Xminlim <= motenc.0.in-00
linksp Xminlim => axis.0.neg-lim-sw-in
linksp Xmaxlim <= motenc.0.in-01
linksp Xmaxlim => axis.0.pos-lim-sw-in
linksp Xhome <= motenc.0.in-02

linksp Xhome => axis.0.home-sw-in
```

Slika 6. Definisane veze ka graničnim prekidačima i sensorima za inicijalizaciju

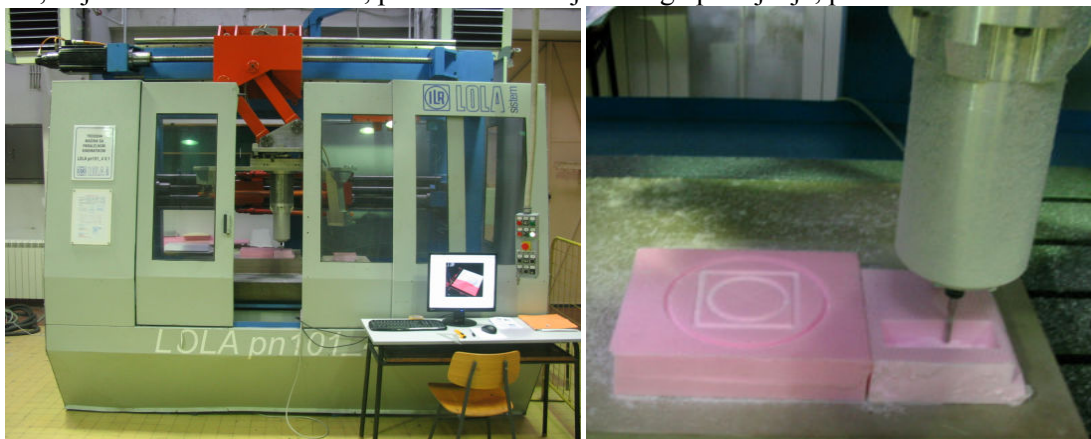
3.5. Konfigurisanje hardvera upravljačke jedinice

Prvi prototip upravljačke jedinice izrađen je na hardverskoj osnovi standardnog PC računara uz korišćenje MOTENC-Lite interfejsa za spregu sa pogonima i mernim sistemima mašine alatke, slika 7.



Slika 7. Upravljačka jedinica otvorene arhitekture za upravljanje mašinom sa paralelnom kinematikom

Konfigurisani upravljački sistem je integrisan u upravljački orman mašine sa paralelnom kinematikom LOLA pn101_4 V.2, slika 8a). Sistem za programiranje koristi CAD/CAM sistem Pro/Engineer WildFire V4. Prvi delovi, koji su obrađeni na mašini, prilikom testiranja novog upravljanja, pokazani su na slici 8b).



a) Nova verzija mašine LOLA pn101_4 V.2

b) Prvi obrađeni probni delovi

Slika 8. Mašina sa paralelnom kinematikom, LOLA pn101_4 V.2, na kojoj je implementirana nova upravljačka jedinica otvorene arhitekture i primer probne obrade

4. ZAKLJUČAK

U radu je predstavljen modularan softverski sistem, prilagodljiv najrazličitijim tipovima mašina alatki i robota različitih kinematičkih struktura. Raznolikost i modularnost je karakteristika koja će biti prepoznata i od strane krajnjih korisnika, rukovaoca, kroz intuitivnost, raznovrsnost i prilagodljivost korisničkih interfejsa. Razvijeni prototip pruža dobru osnovu za nastavak istraživanja na ovom polju i otvara mogućnost da se uz minimalna ulaganja razvije upravljačka jedinica vrhunskih performansi, koja će zbog niske tržišne cene biti van konkurencije na domaćem tržištu.

5. LITERATURA

- [1] EMC - Enhanced Machine controller web site - www.linuxcnc.org
- [2] NIST - National Institute of Standards and Technology web site - <http://www.nist.gov/index.html>
- [3] Real-Time Control Systems Library, Software and Documentation, <http://www.isd.mel.nist.gov/projects/rcslib/>
- [4] Milutinovic D., Glavonjic M., Kvirgic V., Zivanovic S., A New 3-DOF Spatial Parallel Mechanism for Milling Machines with Long X Travel, pp. 345-348, Annals of the Vol54/1, CIRP 2005. doi:10.1016/S0007-8506(07)60119-X
- [5] Glavonjic, M., Milutinovic, D., Zivanovic, S., Dimic, Z., Kvirgic, V., Desktop 3-axis parallel kinematic milling machine, International Journal of Advanced Manufacturing Technology, Volume 46, (2010), pp 51-60 doi: 10.1007/s00170-009-2070-3
- [6] Milutinovic D, Glavonjic M, Slavkovic N, Dimic Z, Zivanovic S, Kokotovic B, Tanovic Lj (2011) Reconfigurable robotic machining system controlled and programmed in a machine tool manner. Int J Adv Manuf Technol., Vol. 53, pp1217-1229, doi:10.1007/s00170-010-2888-8

Zoran Dimić, Miloš Glavonjić, Dragan Milutinović, Saša Živanović,
Vladimir Kvirgić, Marija Milićević

OPEN ARCHITECTURE CONTROL SYSTEM FOR THREE AXES PARALLEL KINEMATICS MILLING MACHINE

Summary

This paper gives a methodology for configuring of open architecture control system for three axes parallel kinematics milling machine. Software and hardware which can be used for control of multi axes machine tool and industrial robots is tested on real machine. The control of servo drives is carried out using PC based platform and Linux operating system.

Key words: control, parallel kinematics machine, servo drives, open architecture control (OAC)