

# POVEZAVA OSEBNEGA RAČUNALNIKA Z ROBOTSКИM KRMILNIKOM ASEA-IRB

INFORMATICA 4/91

**Keywords:** computer link, robot controller,  
personal computer, application protocol

Michele Leonardi, Aleš Klofutar,  
Ivo Gorkič  
Institut Jožef Stefan

Prispevek opisuje povezavo robotskega krmilnika ASEA IRB6/L z osebnim računalnikom. Komunikacija temelji na protokolu asinhronske serijske komunikacije (ADLP10) in aplikacijskem protokolu (ARAP). Prvi del opisuje električno povezavo robotskega krmilnika in osebnega računalnika ter oba protokola. Večji poudarek je na aplikacijskem protokolu, saj ta omogoča dodatke oziroma popravke in dodelave programa brez prevelikih težav. Na podlagi aplikacijskega protokola je nastala knjižnica ukazov. Drugi del vsebuje spisek ukazov in njihovo uporabo ter opisuje delovanje programa *Proto*, ki je nastal hkrati s knjižnico ukazov.

## COMPUTER LINK FOR ASEA ROBOT CONTROLLER

The paper presents the computer link for ASEA IRB robot controller. Communication is based on asynchronous communication protocol (ADLP10) and application protocol (ARAP). Electrical link between the controller and the personal computer together with both protocols are presented in the first part of the paper. The application protocol is more emphasized due to its better possibilities of improvement. A library of instructions was developed on the basis of the application protocol. The list of instructions and the description of program *Proto*, which was developed together with the library are described in the second part of the paper.

## 1. Uvod

Vse večja robotizacija proizvodnih procesov pogojuje potrebo po usklajenem delovanju robotov in njihovem prilagajanju okolici. Robotski krmilniki poskrbijo, da robot opravi vnaprej zadane naloge, katerih obsežnost je omejena z zmogljivostmi robotskega krmilnika. Zmogljivosti robotskega krmilnika lahko bistveno povečamo s povezavo z zmogljivejšim osebnim računalnikom.

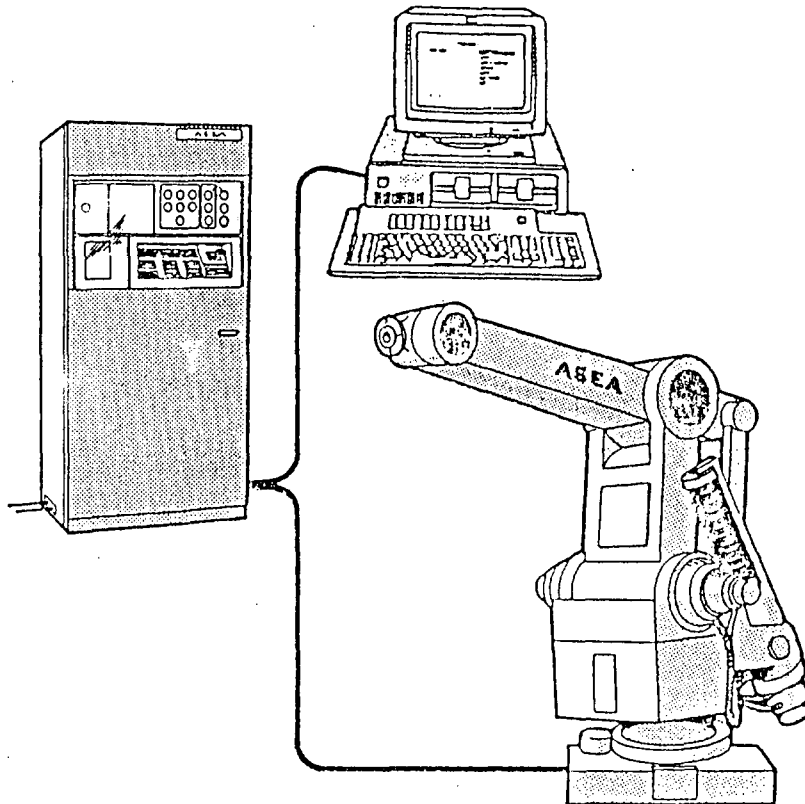
V članku smo opisali povezavo robotskega krmilnika ASEA IRB6/L z računalnikom PC AT. Povezava omogoča izvajanje večine ukazov, ki jih lahko izvedemo neposredno z robotskega krmilnika tudi z osebnega računalnika.

## 2. Komunikacijski protokol

Povezava med robotskim krmilnikom in osebnim računalnikom je omogočena s protokoloma, ki jih je predpisala ASEA: ADLP 10 (ASEA Data Link Protocol) [2] in ARAP (ABB Robot Application Protocol) [1].

### 2.1 Električna povezava

Na strani robotskega krmilnika smo uporabili poseben komunikacijski vmesnik DSCA114 in pri njem ustrezno nastavili vhodno/izhodni naslov vmesnika in prekinitveni nivo. Na strani osebnega računalnika pa smo uporabili že vgrajeni vhodno/izhodni vmesnik z ustrezno nastavitvijo.



Slika 1: Povezava robotskega krmilnika z osebnim računalnikom

## 2.2 ADLP10

ADLP-10 je postopek za asinhronsko komunikacijo med dvema postajama v hierarhičnem sistemu. Postopek temelji na protokolih ECMA-16 in ECMA-24.

Prenos podatkov je asinhronski, serijski in pol-dvosmeren. Lahko je iniciiran z obeh strani. Vsak znak (podatkovni, kontrolni) je sestavljen iz 8 bitov + parnostni bit + stop bit. Preverjanje pravilnosti komunikacije se vrši z vertikalno in horizontalno kontrolo parnosti.

Za komunikacijo prek serijskega vhodno/izhodnega vmesnika smo razvili podprogram, ki omogoča odpiranje in zapiranje komunikacijskega kanala ter oddajo in sprejem posameznih znakov. Pri odpiranju komunikacijskega kanala je potrebno specificirati številko komunikacijskega kanala (COM1, COM2), hitrost prenosa (npr. 9600 b/s), število bitov na znak, število stop bitov in parnost (npr. liha parnost). Podprogram poleg tega omogoča hkraten sprejem in oddajo znakov ter detektira napake med sprejemom.

Izmenjavo informacij med postajama razdelimo v tri faze

1. Vzpostavitev komunikacije
2. Prenos informacij
3. Zaključek komunikacije

Prenos informacij poteka med drugo fazo. Za spremembo smeri komunikacije moramo izmenjavo zaključiti in ponovno vzpostaviti komunikacijo.

Postajo, ki vzpostavi komunikacijo imenujemo master, podrejeno pa slave. Master je zadolžen za prehajanje med fazami in vzdrževanje komunikacije.

## 2.3 ARAP

Komunikacija med postajama poteka prek sporočil, ki so definirana z aplikacijskim protokolom (ARAP). Sporočilo predstavlja informacije, ki so med seboj logično povezane in sestavljajo enoto. Sestavljeno je iz enega ali več telegramov.

### 2.3.1 Telegrami

Informacijo, ki se izmenjuje med robotskim krmilnikom in osebnim računalnikom, razdelimo v manjše enote, ki jim pravimo telegrami. Vsak telegram je sestavljen iz glave in podatkovnega polja. Glava je vedno enake velikosti (8 bytov) in vsebuje informacijo, ki temelji na aplikacijskem protokolu (velikost telegrama, funkcijsko kodo, tip telegrama, indikator dolžine telegrama, status odgovora, naslov pošiljatelja in naslov prejemnika telegrama).

Funkcijska koda določa namen sporočila. V dokumentaciji je predvidenih 44 različnih funkcijskih kod.

Imamo dva tipa podatkovnih polj (koda napake, podatkovno polje). Velikost in vsebina podatkovnega polja se spreminjata v odvisnosti od tipa sporočila in je v določenih primerih tudi izpuščena.

Obstajajo trije različni tipi telegramov:

**Ukaz** lahko pošljeta tako robotski krmilnik kot osebni računalnik in pomeni, da pošiljatelj zahteva, da sprejemnik izvrši neko nalogo. Ukaz se lahko sestoji le iz glave ali pa iz glave in podatkovnega polja. Podatkovno polje je lahko daljše od maksimalne dolžine telegrama, zato se za ukaz lahko uporabi več povezanih telegramov. Ukaz vedno pričakuje odgovor (pozitiven oz. negativen).

**Odgovor** je poslan le kot potrdilo ukaza in ga lahko pošljeta tako robotski krmilnik kot osebni računalnik. Tako kot ukaz se tudi odgovor lahko sestoji le iz glave ali pa iz glave in podatkovnega polja. Če je podatkovno polje daljše od maksimalne dolžine telegrama, se za odgovor lahko uporabi več med seboj povezanih telegramov. Lahko ga uporabimo tako za pozitivno kot tudi negativno potrditev ukaza. V primeru negativne potrditve se podatkovno polje nadomesti s 16 bitno kodo napake. Če za neko funkcijsko kodo podatkovno polje ne obstaja, se telegramu doda 16 bitna koda napake v primeru negativnega odgovora.

**Spontano sporočilo** lahko pošlje le robotski krmilnik računalniku in ni pogojeno z ukazom. Robotski krmilnik pošilja spontana sporočila kot posledico različnih dogodkov v delovanju

robotu npr. izklop v sili, sistemske napake itd.

## 3. Knjižnica ukazov

Na osnovi aplikacijskega protokola smo v programskem jeziku Turbo Pascal razvili knjižnico ukazov, ki omogočajo vodenje in programiranje robotskega krmilnika z osebnega računalnika. Osnovo predstavljajo ukazi, ki so z določenimi funkcijskimi kodami direktno dosegljivi iz aplikacijskega protokola.

Vsi ukazi so oblike `Exec_...` in kot prvi parameter vedno vrnejo kodo napake (`ErrResult`), ki jo javi robotski krmilnik. Če je vrednost `ErrResult = 0`, pomeni, da je prišlo do prekinitve komunikacije, če pa `ErrResult = -1`, pa da je robotski krmilnik ukaz izvršil. Ostale vrednosti kode napake pa najdemo v dokumentaciji.

Ukaze delimo na osem skupin.

**Statusni ukazi** omogočajo branje pozicije in orientacije robotu ter branje stanja motorjev robotu.

**Ukazi povezani z robotskimi programi** omogočajo prenos programov z računalnika na robotski krmilnik, brisanje programov iz spomina krmilnika, zagon in ustavitev programov iz krmilnika.

**Ukazi povezani s premikanjem robotu** Za premik robotu moramo poslati tri ukaze

1. Začetna točka
2. Ukaz za absolutni ali relativni premik
3. Končna točka

Ukaz omogoča nastavitev hitrosti robotu in absolutno oziroma relativno premikanje robotu.

**Ukazi povezani s koordinatami orodja** omogočajo vpis, brisanje in branje koordinat orodja ter aktiviranje določenih koordinat.

**Ukazi povezani z registri** omogočajo vpisovanje in branje iz

- registrov
- lokacijskih registrov
- digitalnih izhodov

- digitalnih vhodov
- frame registrov

Ukazi povezani s konfiguracijo omogočajo branje in vpisovanje konfiguracije na robotski krmilnik.

#### Ukazi povezani z Vision sistemom

omogočajo branje podatkov, ki so povezani z robotskim vidom s krmilnika in lociranje modelov, ki so v vidnem polju kamere. seveda jih lahko uporabljamo le, če je krmilnik opremljen z Vision sistemom.

#### Ukazi povezani s prijemalko robota

omogočajo zapiranje in odpiranje prijemalke robota.

## 4. Program Proto

Na podlagi knjižnice ukazov smo razvili program Proto, in z njim preverili delovanje knjižnice ukazov. Poleg osnovnih ukazov, ki smo jih predstavili v prejšnjem poglavju, vsebuje program še nekaj kombinacij le teh, zraven pa še omogoča sprejem ukazov s strani robotskega krmilnika in sprejem spontanih sporočil (Spontaneous messages) s strani robotskega krmilnika.

### 4.1 Podatki o stanju robota

Program periodično zahteva od robotskega krmilnika podatke o stanju, v katerem se robot nahaja. Zgornja vrstica je rezervirana za sporočila uporabniku. Tu se izpisujejo naslednje stvari

- Stanje pripravljenosti
- Vsebina spontanih sporočil [1, str.78]
- V primeru nepravilno sprejetega ukaza s strani robotskega krmilnika, sporočilo, ki ga definira koda napake [1, str.81]
- Potrditev uspešno izvedenega ukaza.

V drugi vrstici se izpisuje stanje motorjev robota (Stand by, Operate, Execute, Emergency stop) ter program in vrstica programa, v kateri smo. V tretji vrstici imamo stanje enote za programiranje (connected, disconnected). V četrti pa pozicija (v mm) in orientacija (v stopinjah) robota.

Spodnji dve tretjini ekrana sta rezervirani za ukaze uporabnika.

## 4.2 Ukazi

Ukazi so razdeljeni na pet sklopov

1. Ukazi povezani s programiranjem robotskega krmilnika (**Robot program commands**)  
Uporabljajo ukaze povezane z robotskimi programi iz knjižnice ukazov poleg tega pa še izpis programov na ekran in tiskalnik.
2. Branje podatkov z robotskega krmilnika (**Data reading from SC**)  
Omogočajo branje podatkov iz posameznih registrov robotskega krmilnika, branje konfiguracije in koordinat orodja.
3. Vpisovanje podatkov na robotski krmilnik (**Data writing from SC**)  
Omogočajo vpis podatkov iz posameznih registrov robotskega krmilnika, vpis konfiguracije in koordinat orodja.
4. Ukazi povezani z robotskim vidom (**Vision system data**)  
Uporabljajo ukaze povezane z robotskim vidom iz knjižnice ukazov.
5. Neposredni ukazi (**Direct commands**)  
Omogočajo aktiviranje določenih koordinat orodja in določenega koordinatnega sistema odpiranje in zapiranje prijemalke, sinhronizacijo, vžig in izklop motorjev robota ter premikanje robota. Premikanje robota je mogoče
  - absolutno ali relativno: podamo koordinate pozicije (v mm) in orientacije (v stopinjah) robota.
  - zvezno: s pritiskom na določeno tipko se robot premakne v smeri, ki jo določa tipka.

## 5. Zaključek

Programski paket omogoča povezavo robotskega krmilnika ASEA z osebnim računalnikom.

Trenutno stanje robota in njegove okolice lahko izčrpnje in hkrati bolj pregledno prikažemo na

```

message:  ABB computer link at your service

Op.mode  STAND BY      Program  0/10

Programming unit  CONNECTED

x  471.5  y  0.0  z  500.3  roll  41.3  pitch  -77.1

1 : Robot program commands
2 : Data writing from SC
3 : Data reading from SC
4 : Vision system data
5 : Direct commands

```

Slika 2: Program Proto

zaslonu osebnega računalnika kot na dvovrstičnem prikazovalniku robotove prenosne ukazne plošče. S tem je mogoče statistično spremljanje in iskanje optimalnih časov delovnih faz.

Pomembna je tudi možnost pisanja programov za robotski krmilnik na osebem računalniku. S tem prenesemo programiranje iz proizvodnih dvoran v človeku prijaznejše okolje in omogočimo simulacijo delovanja robota na osebem računalniku. Omogočena nam je tudi uporaba vseh programskih orodij namenjenih za izdelavo programov, kar nam v veliki meri olajša pisanje le teh.

Ob vseh drugih prednostih je zelo pomembna tudi možnost implementacije nove programske opreme. Robotski krmilnik ASEA namreč predstavlja zaprt sistem in implementacija nove programske opreme je mogoča le prek povezave z osebnim računalnikom.

Programski paket je zgrajen modularno in omogoča lahko vključevanje knjižnice ukazov v programe napisane v programskem jeziku Turbo Pascal. Omogočena je kontrola napak, saj rutine kot enega izhodov vračajo kodo napake.

Program *Proto*, ki smo ga razvili za preizkus

delovanja knjižnice ukazov poleg osnovnih funkcij omogoča še izpis robotskih programov na ekran in tiskalnik.

Programski paket bi lahko uporabili tudi za uskladitev delovanja več robotov. Protokol to omogoča, seveda pa so potrebne tudi ustrezne dodelave programskega paketa in pa algoritmi, ki bi koordinirali sodelovanje več robotov.

## Literatura

- [1] ABB Robot Application Protocol (ARAP) for Computer Link ABB, Sept. 1988
- [2] ASEA Asynchronous Communication for Computer Link-ADLP 10, Jan. 1984
- [3] ABB Industrial Robot System IRB6/2, July 1983
- [4] Ivo Gorkič: Program za povezavo industrijskega robota z osebnim računalnikom, Diplomaska naloga, Fakulteta za elektrotehniko in računalništvo, Ljubljana, Junij 1991