

## POSTUPAK REALIZACIJE INSTRUMENTACIONOG DRAJVERA ZA PC KOMPATIBILNU BROJAČKU KARTICU

Vujo Drndarević, Nenad Jevtić, *Saobraćajni fakultet, Beograd*  
Biljana Veselinović, *Institut Vinča, Beograd*

**Sadržaj** – U radu je opisan postupak realizacije instrumentacionog drajvera za LabView programsko okruženje. Na bazi iznetih principa i izvršenih analiza realizovan je instrumentacioni drajver za PC kompatibilnu brojačku karticu. Na taj način došlo se do osnove sistema za prikupljanje ili generisanje vremenskih signala na bazi personalnog računara.

### 1. UVOD

Instrumentacioni drajver se sastoji iz skupa programskih rutina koje upravljaju funkcijama programabilnog instrumenta. Pod programabilnim instrumentom podrazumeva se instrument povezan sa PC računarom preko standardnog komunikacionog interfejsa (RS-232, IEEE-488, USB i dr.) ili akviziciona kartica direktno povezana na zajedničku magistralu računara.

Korišćenjem instrumentacionih drajvera značajno se olakšava rešavanje problema softverske podrške pri realizaciji računarski podržanog merno-upravljačkog sistema. S druge strane, kada se poseduje instrumentacioni drajver za određeni uređaj ili instrument, onda za uspešnu realizaciju programa aplikacije nije neophodno detaljno poznavanje načina funkcionisanja hardvera uređaja tj. programskog protokola po kome se ovo funkcionisanje odvija. To znači da se od programera koji razvijaju program aplikacije ne zahteva detaljno poznavanje organizacije i načina funkcionisanja hardvera, čime se značajno može povećati efikasnost u radu.

Od instrumentacionog drajvera se zahteva da ima potpunu kontrolu nad hardverom. Svaka programska rutina instrumentacionog drajvera obavlja određenu funkciju kao što je konfigurisanje, upis, očitavanje, aktiviranje (trigerovanje) i dr.

U ovom radu dat je postupak izrade instrumentacionog drajvera za PC kompatibilnu brojačku karticu koja se koristi za brojanje impulsa ili generisanje vremenskih signala. Drajver je prilagođen za primenu u LabView programskom okruženju i dat je u vidu virtuelnog instrumenta [1,2]. Na ovaj način obezbeđena je programska podrška ranije razvijenoj PC kompatibilnoj brojačkoj kartici [3], čime se krajnjem korisniku omogućuje da na veoma efikasan način, u grafičkom okruženju koje pruža LabView, kreira sopstvenu aplikaciju za specifičan merno-upravljački sistem.

### 2. ZAHTEVI

Jedan od najvažnijih zahteva koji se postavlja pred instrumentacioni drajver je da omogući potpunu kontrolu hardvera. Imajući u vidu činjenicu da savremeni elektronski uređaji i instrumenti raspoložu velikim brojem funkcija i načina rada, pri razvoju instrumentacionog drajvera

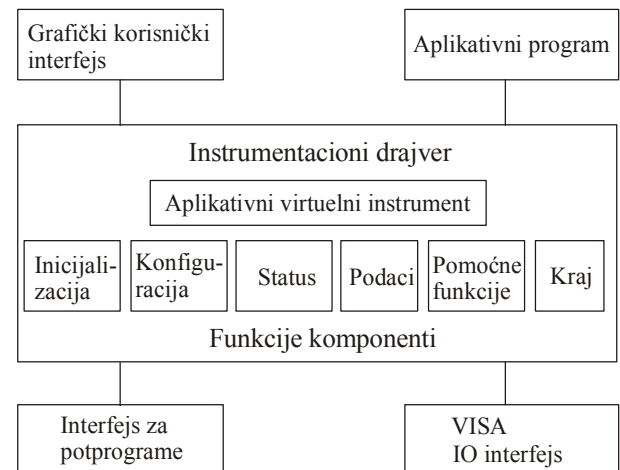
konzistentno se treba držati standardnog modela instrumentacionog drajvera koji počiva na modularnoj strukturi.

Pri projektovanju instrumentacionog drajvera potrebno je poštovati neke od osnovni principa. Da bi instrumentacioni drajver programski obavljao određene funkcije on mora biti tako koncipiran da se pri njegovom izvršavanju ne zahteva interakcija korisnika. Ovaj zahtev ne isključuje mogućnost da korisnik, preko odgovarajućeg grafičkog korisničkog interfejsa, unese željene parametre vezane za rad uređaja ili za obavljanje komunikacionih funkcija.

Da bi se omogućilo jednovremeno korišćenje istog instrumentacionog drajvera od strane većeg broja identičnih uređaja, unutar instrumentacionog drajvera ne bi trebalo vršiti memorisanje podataka. U protivnom, podaci memorisani od strane jednog uređaja mogli bi biti izmenjeni ili pogrešno interpretirani od strane drugog uređaja.

### 3. ORGANIZACIJA INSTRUMENTACIONOG DRAJVERA

Instrumentacioni drajver predviđen za primenu u LabView programskom okruženju sastoji se od softverskih modula ili virtuelnih instrumenata, što je uobičajeni naziv za programske celine u grafičkim programskim jezicima. Ovim programskim celinama korisnik može direktno pristupiti ili ih pozivati iz programa aplikacije. Na slici 1 prikazan je opšti model instrumentacionog drajvera.



Sl. 1 Opšti model instrumentacionog drajvera

Model sa slike 1 pokazuje vezu instrumentacionog drajvera sa programom aplikacije, internu organizaciju programskih modula instrumentacionog drajvera i vezu instrumentacionog drajvera sa programskim rutinama i hardverom instrumenta ili uređaja.

Instrumentacioni drajver sastoji se od funkcionalne osnove ili tela, programskog interfejsa, interaktivnog korisničkog interfejsa, VISA IO interfejsa i interfejsa sa drugim programskim rutinama.

Funkcionalna osnova drajvera u suštini sadrži programski kod drajvera. Preko programskog interfejsa drajver se poziva od strane programa aplikacije. Korisniku se daje mogućnost da preko interaktivnog grafičkog korisničkog interfejsa bliže upozna funkcije instrumentacionog drajvera.

VISA (*Virtual Instrument Software Architecture*) IO interfejs predstavlja standardni instrumentacioni interfejs za komunikaciju sa perifernim uređajima i instrumentima. Ova komunikacije se može obavljati putem standardnih komunikacionih interfejsa (IEEE-488, RS-232 i sl.) ili preko registara, što se često koristi kada je periferija direktno povezana na računarsku magistralu. Za izvršenje određenih zadataka, kao što je npr. generisanje poruke o grešci, drajver može pozvati odgovarajuću programsku rutinu, što se definiše preko odgovarajućeg mehanizma.

Funkcionalna osnova ili telo drajvera sadrži veći broj programskih modula, pri čemu svaki od ovih modula ima specifičan zadatak. U okviru tela drajvera vrši se inicijalizacija i konfiguracija hardvera, očitavanje statusa, očitavanje podataka i obavljanje drugih funkcija. Jedna od važnih funkcija je i završetak rada programa instrumentacionog drajvera, što dovodi do prekida veze programa sa hardverom, pa samim tim i do oslobađanja korišćenih računarskih resursa.

Funkcionalna osnova ili telo LabView instrumentacionog drajvera sadrži dve osnovne kategorije virtuelnih instrumentata: skup komponenti i aplikativne virtuelne instrumente. Komponente upravljaju specifičnim funkcijama instrumenta a aplikativni virtuelni instrumenti, korišćenjem ovih komponenti, omogućuju merne i test funkcije instrumenta. U određenim primenama aplikativni virtuelni instrument predstavlja jedini instrumentacioni drajver koji se koristi za upravljanje hardverom.

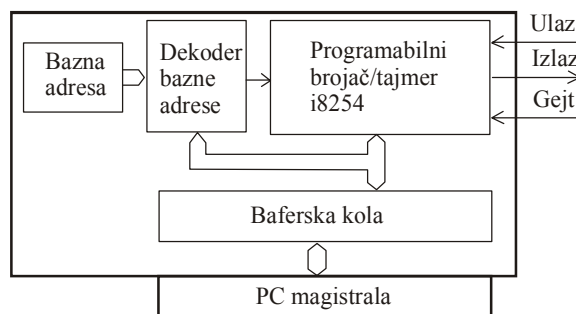
#### 4. REALIZACIJA INSTRUMENTACIONOG DRAJVERA ZA BROJAČKU KARTICU

Put u razvoju instrumentacionog drajvera koji omogućuje potpunu kontrolu hardvera vodi preko detaljnog poznavanje organizacije hardvera odnosno instrumenta. Takođe, pri koncipiranju funkcija instrumentacionog drajvera potrebno je posedovati iskustvo u korišćenju hardvera odnosno uređaja kojim se upravlja. Da bi se došlo do ovih znanja poželjno je, pored korišćenja tehničke dokumentacije koja prati određeni hardver, steći iskustvo kroz praktičan rad i korišćenje uređaja ili instrumenta u različitim konfiguracijama. Posebnu pažnju treba pokloniti delovima uputstva koji se odnose na programiranje instrumenta i skup instrukcija koje se koriste u obezbeđenju mernih i upravljačkih funkcija. Na osnovu ovakvih analiza i stečenih iskustava u radu sa hardverom može se izvršiti grupisanje modula i zadataka što predstavlja osnovu za uspostavljanje modularne i hijerarhijski organizovane strukture drajvera.

Postupak realizacije instrumentacionog drajvera biće dat na primeru PC kompatibilne brojačke kartice (slika 2) [3]. Ova kartica može se koristiti za brojanje impulsa ili generisanje odgovarajućih vremenskih signala. Za realizaciju ovih funkcija koristi se programabilni brojač/tajmer i8254

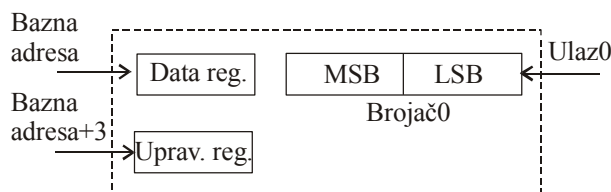
[4]. Prema navedeno programabilno kolo sadrži tri 16-bitna brojača i može da obavlja šest različitih funkcija, ovde će pažnja biti posvećena samo jednoj, brojanju impulsa, a za ovu funkciju biće korišćen samo jedan 16-bitni brojač. Redukovanje broja posmatranih funkcija može se dovesti u vezu sa dozvoljenim obimom izlaganja, uz nameru da iznete analize ne izgube svoju praktičnost.

Na ulaz programabilnog brojača se, posredstvom interfejsnog kola dovode impulsi koji se broje. Brojanje impulsa je moguće jedino kada je na liniji gejt visok naponski nivo. Generisanje izlaznih signala vrši se preko linije izlaz.



Sl.2 Blok šema PC kompatibilne brojačke kartice

Uprošćen model programabilnog kola i8254 kada se koristi samo jedan 16-bitni brojač prikazano je na slici 3.



Sl. 3 Uprošćen model programabilnog brojača/tajmera i8254

Brojačka kartica je mapirana u IO adresnom prostoru i zauzima četiri IO adrese, pri čemu prve tri IO adrese zauzimaju registri preko kojih se pristupa odgovarajućim brojačima a četvrtu upravljački registar.

Konfigurisanje programabilnog brojača tj. izbor načina rada kola vrši se upisom odgovarajućeg sadržaja u upravljački registar. Inicijalizacija brojača vrši se u dva koraka, upisom početnog dvobajtnog sadržaja brojača. Upis se vrši preko bazne adrese koja je pridružena odgovarajućem brojaču. Nakon inicijalizacije, programabilni brojač i8254 broji unazad.

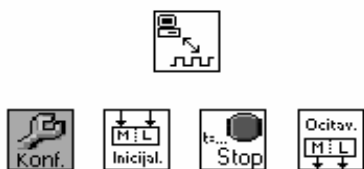
Očitavanje sadržaja brojača vrši se takođe u dva koraka, preko odgovarajuće bazne adrese. Pre očitavanja, sadržaj brojača potrebno je lečovati. Na taj način izbegava se mogućnost da se sadržaj promeni tokom same operacije očitavanja. Posle izdavanja komande za lečovanje očitava se status i lečovani sadržaj brojača. Postupak konfiguracije, inicijalizacije i očitavanja brojača i izračunavanje odbroja sistematizovan je u tabeli 1.

Prema sekvencama datim u tabeli 1 realizovane su komponente instrumentacionog drajvera brojačke kartice. Korišćenjem ovih komponenti realizovan je aplikativni virtuelni instrument koji omogućuje kompletnu brojačku funkciju kartice.

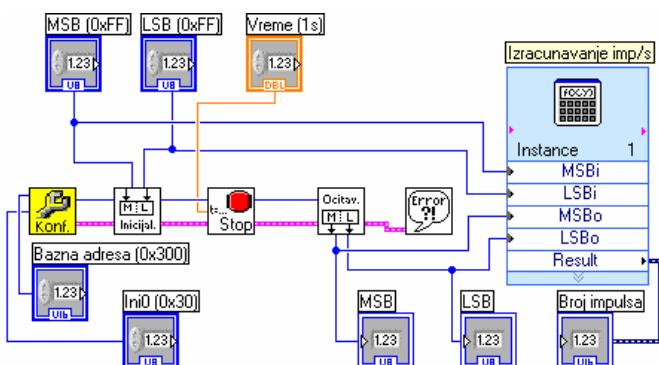
Tabela 1. Postupak programiranja brojača

	Operacija	Sadržaj	Adresa	Komentar
1	Upis	30h	Bazna+3	Inicijalizacija brojača0
2	Upis	FFh	Bazna	Početni sadržaj brojača0 LSB
3	Upis	FFh	Bazna	Početni sadržaj brojača0 MSB
4	Merenje vremena, 1000 ms			
5	Upis	C2h	Bazna+3	Zamrzavanje sadržaja brojača0
6	Očitavanje		Bazna	Očitavanje statusa brojača0 b <sub>7</sub> b <sub>6</sub> ...b <sub>0</sub>
7	Ako je b <sub>7</sub> b <sub>6</sub> = 10 <sub>2</sub> došlo je do prekoračenja brojača0 - signalizirati status			
8	Očitavanje		Bazna	Očitavanje sadržaja brojača0 na kraju intervala merenja LSB
9	Očitavanje		Bazna	Očitavanje sadržaja brojača0 na kraju intervala merenja MSB
10	Izračunavanje odbroja $Br = INIC - (MSB * 2^8 + LSB) +$			INIC je početni sadržaj brojača

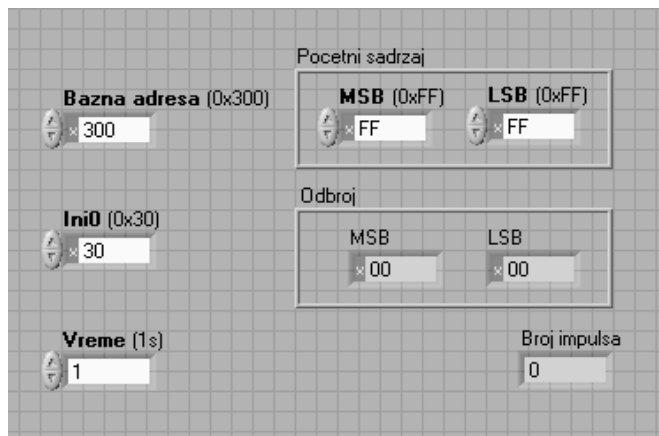
Na slici 4 data je blok šema instrumentacionog drajvera a na slici 5 dijagram i grafički korisnički interfejs aplikativnog virtuelnog instrumenta.



Sl. 4 Blok šema instrumentacionog drajvera brojačke kartice



Sl. 5a Dijagram aplikativnog virtuelnog instrumenta



Sl. 5b Grafički korisnički interfejs aplikativnog virtuelnog instrumenta

### 5. ZAKLJUČAK

U radu su analizirani principi i dat je postupak razvoja instrumentacionog drajvera. Izvršene analize praktično su demonstrirane na primeru realizacije instrumentacionog drajvera za PC kompatibilnu brojačku karticu za rad u LabView programskom okruženju.

Pored osnovne namere, da se ukaže na funkcije, organizaciju i put razvoja instrumentacionog drajvera, došlo se do jednog praktičnog rezultata koji se, sa posmatranom brojačkom karticom, može uspešno primeniti u brojanju impulsa koji potiču iz generatora signala ili raznovrsnih senzora kao što su turbinski senzor protoka, impulsni detektor zračenja i dr. Naravno, performanse ovakvih mernih sistema određene su kako karakteristikama hardvera tako i karakteristikama karakteristika softvera, posebno operativnog sistema PC računara, ako se zahteva determinisan odziv sistema.

### LITERATURA

- [1] National Instruments, www.ni.com
- [2] V. Drndarević, *Personalni računari u sistemima merenja i upravljanja*, Akademska misao, Beograd, 2003.
- [3] V. Drndarevic, M. Bolic, S. Spasojevic, B. Samardzic, "A PC-Based Radiation Monitor", *Instruments and Experimental Techniques*, Vol. 42, No.1, 1999., pp. 103-106.
- [4] Intel Corporation, *Microprocessor and Peripheral Handbook*, Vol. I-II, Santa Clara, CA, 1993.

**Abstract** - Method of the realization of the instrument driver for the LabView development tool has been described in this paper. Instrument driver for the PC counting board has been completed based on presented principles and implemented analyses. This was the method for defining the system for the acquisition or generating of timing signals using personal computers.

### DEVELOPING INSTRUMENT DRIVER FOR THE PC COMPATIBLE COUNTING BOARD

Vujo Drndarević, Nenad Jevtić, Biljana Veselinović