

V. Bonačić

automatika

SEPARAT

Optička komunikacija s »ON-LINE« računalom pomoću svjetlosnog pera

DK 681.14—523.8
 IFAC IA 4.5.1

Konstruiran je sistem s računalom PDP-8 koji omogućuje komunikaciju čovjek — računalu u oba smjera pomoću slike.

Rezultati iz računala prebacuju se na ekran osciloskopa. Dobivena slika može sadržavati do 16.192 tačaka. Operator pomoću »svjetlosnog pera« može proizvoljno crtati slike na ekranu. Digitalizirane slike su zapamćene u memoriji i na njima se mogu vršiti logičke i matematske operacije. Sis'em se koristi za kontrolu eksperimenta odnosno procesa. Svjetlosno pero je izvedeno pomoću foto osjetljive diode i »field-effect« tranzistora i šalje signale direktno u računalu kroz »prekidni ulaz«.

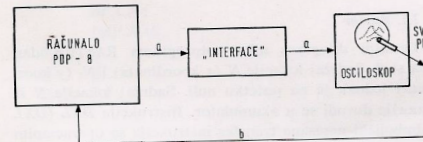
Optical Communication with »ON-LINE« Computer by Light-Pen. A system with the computer PDP-8 has been constructed which makes possible the communication man — computer in both directions by display.

Computer data are displayed on the oscilloscope screen. The display obtained can contain up to 16.192 points. Using a light pen the operator can arbitrarily display pictures on the screen. Digitalized pictures are stored in the memory, and logical and mathematical operations can be performed on them. The system is employed for control of experiments or processes. The light pen is built with a photo-sensitive diode and a field-effect transistor, and is connected directly to the computer interrupt input.

1. Uvod

Za bolje razumijevanje članka sugerira se čitaocu da pročita »Digitalna računala u mjerenju i regulaciji« u istom časopisu, jer autori polaze od pretpostavke da su čitaoci upoznati s terminologijom i elementarnim operacijama u računalu.

Opisan je i analiziran izvedeni sistem* koji omogućuje komunikaciju čovjek — računalu (sl. 1). Podaci koji se na-



Sl. 1. Komunikacija s računalom pomoću osciloskopa i svjetlosnog pera

laze u memoriji računala kao rezultat nekog eksperimenta ili proračuna prebacuju se na ekran osciloskopa preko »interfacea« a pod kontrolom programa (software). Taj put označen je slovom »a«. Operator može pomoću svjetlosnog pera već postojeću sliku na ekranu modificirati brišući odnosno crtajući pojedine dijelove, ili crtajući sliku na praznom ekranu. Operator time istovremeno automatski preko linije b mijenja sadržaj dijela memorije koji je predviđen za podatke.

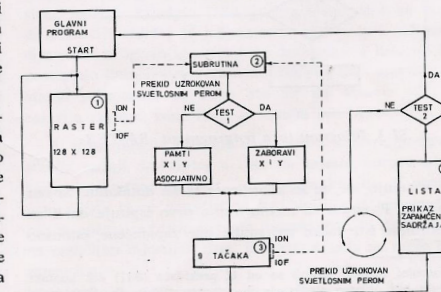
Tako dobivena slika koja je ograničena na 2048 različitih podataka može se preko periferne jedinice bušača prenijeti na perforiranu traku zbog daljnje obrade odnosno fotografirati zbog dokumentacije. Podatke koji se nalaze na perforiranoj traci moguće je učitati u računalu. Jednake deskriptore (podatke) kao i podatke nakon 2048-og računala ne registrira. Program za bušenje i čitanje nije sastavni dio OKR (optička komunikacija s računalom) te ne će biti razmatran. Startanjem programa OKR na

ekranu se pojavljuje pravokutni raster, sl. 8 a, koji sadrži 16 192 (128 × 128) svijetlih tačkica i predstavlja bazu za crtanje slike. Raster se generira kratkim strojnim programom koji sadrži desetak instrukcija. Digitalizirana zapamćena slika sadrži 2048 podataka (svijetlih tačkica) koji su proizvoljno odabrani dio rastera.

Između računala i ekrana katodne cijevi nalazi se »interface« koji pod kontrolom programa OKR omogućuje transfer podataka iz računala na ekran katodne cijevi uz digitalno analognu pretvorbu. »Interface« sadrži registre RP_x i RP_y za podatke odnosno koordinate x i y . Instrukcije DXL i DYL preko selektora koji su sastavni dijelovi »interfacea« prebacuju sadržaj akumulatora na registre podataka koji preko digitalno analognih konvertera otklanjaju zraku proporcionalno veličini koordinate x i y . Instrukcija DIY registrira položaj otklonjene zrake impulsom za osvjjetljenje.

2. Program optičke komunikacije s računalom

Dijagram toka programa OKR prikazan je na sl. 2. Program se sastoji iz četiri potprograma označenih brojevima 1—4. Potprogrami 1, 3, 4 počinju i završavaju u



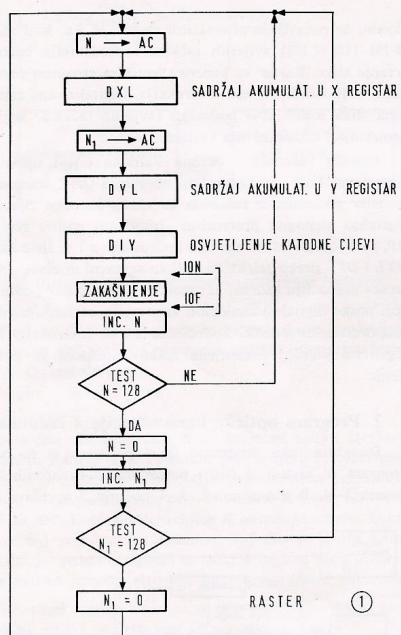
Sl. 2. Dijagram toka programa za optičku komunikaciju s računalom

* Sistem za komunikaciju čovjek — računalu razvijen je i izveden u Institutu »Ruder Bošković«.

glavnom programu što nije zbog jednostavnosti crteža naznačeno. U glavnom programu također je sadržan potprogram koji služi za brisanje dijela memorije rezerviranog za deskriptore (podatke). Takav način pisanja programa nije neophodan ali je praktičan kod uhođavanja tek napisanog programa u kojem se uvijek mogu naći logičke greške kao i one nastale omaškom.

Nakon starta programa OKR i brisanja starih deskriptora program ostaje u petlji RASTERA generirajući sukcesivno tačku po tačku šaljući ih na ekran katodne cijevi preko interfeasa. Zbog brzine računala i pamćenja fosfora katodne cijevi vidi se cijeli raster.

Ako se bilo koja svijetla tačkica dotakne svjetlosnim perom SP, svjetlosni impuls se preko foto-diode i pojačala SP-a pretvori u električni impuls i postavi zahtjev za prekid programa (program interrupt). Ako je prekid programa omogućen instrukcijom ION (»interrupt on«), nastaje skok u subrutinu (prekidni program), odnosno program se nastavlja izvan petlje rastera. TEST 1 koji se nalazi u subrutini ispituje da li je preklopnik ϕ konzolnog [»switch«] registra* u stanju »1«. Normalno prekidač se na-



Sl. 3. Dijagram toka potprograma »RASTER«

lazi u stanju »0«, te se koordinate tačke dotaknute SP-om zapamte. Program za memoriranje prvo ispituje da li su koordinate iste tačke već ranije bile zapamćene, odnosno

* Konzolni registar sastoji se od 12 prekidača (0-11) čiji kontakt može biti otvoren stanje »1« ili zatvoren stanje »0«. Preklopnici se ručno namještaju prije ili za vrijeme toka programa a program, ukoliko to programer želi, ispituje stanje konzolnog registra, odnosno bilo kojeg preklopnika.

da li postoji slobodno mjesto u dijelu memorije koji je predviđen za sadržaj. Ako se preklopnik ϕ nalazi u stanju »1« koordinate dotaknute tačke se zaboravljaju, odnosno brišu ako su ranije bile zapisane.

Po završetku pamćenja ili brisanja SP-om odabranih koordinata, modificira se sadržaj nulte lokacije memorije tako da se iz subrutine nakon izvršenja instrukcije JMP 1 ϕ program ne nastavlja na mjestu u rasteru gdje je nastao prekid već glavni program prelazi na potprogram 9 TACAKA.

Potprogram »9 TACAKA« generira oko upravo zapamćene ili izbrisane koordinate x_{ij} vijenac od osam na rasteru susjednih tačaka. Program zatim skače preko glavnog programa na potprogram LISTA koji omogućuje prikaz zapamćenog sadržaja. TEST 2 testira konzolni registar (preklopnik 1). Ako se preklopnik 1 nalazi u stanju 1, program skače natrag na RASTER očekujući novi interrupt. Ako je preklopnik 1 u stanju 0 program se nastavlja u petlji »9 TACAKA — LISTA«.

Ako se svjetlosno pero pomakne u bilo kojem smjeru od središnje tačke x_{ij} nastaje prekid, a vijenac nastaje oko nove tačke x_{ij} . Na ekranu katodne cijevi vijenac se nalazi oko tačke koja je posljednja izazvala prekid i putuje u smjeru pomicanja SP-a ostavljajući tačkasti svijetli trag, dok su koordinate svijetlih tačkica zapamćene u memoriji računala. Diskontinuitet u crtanju linije moguće je postići sa preklopnicima 0 i 1 konzolnog registra. Sa preklopnikom ϕ u stanju »1« započinje operacija brisanja sa SP ne ostavlja trag, a sa preklopnikom 1 u stanju 1 na ekranu se pojavljuje raster pa se sa SP može započeti crtanje od bilo koje druge tačke rastera.

2.1. Raster

Detaljni dijagram toka potprograma RASTER dan je na sl. 3. Sadržaj lokacije N (x koordinata) i N_1 (y koordinata) jednak je na početku nuli. Sadržaj lokacije N iz memorije dovodi se u akumulator. Instrukcija DXL (DXL je simbolički napisana transfer instrukcija sa operacionim kodom 6 i adresnim dijelom kojim selektira selektor X) resetira RP_x (registar za podatak x) i neposredno nakon toga izjednači sadržaj akumulatora i RP_x . Digitalni podatak sa RP_x u digitalno analognom konvertoru pretvori se u napon koji otklanja zraku osciloskopa u x smjeru proporcionalno sadržaju lokacije N . U očišćeni akumulator dovodi se sadržaj lokacije N_1 i nakon izvršenja instrukcije DYL zraka osciloskopa se otkloni u y smjeru proporcionalno sadržaju lokacije N_1 . Instrukcija DIY proizvede impuls za osvijetljenje koji registrira položaj otklonjene zrake. Ako se u tom trenutku na ekranu ispred svijetle tačke nalazila fotodiode SP-a u računalo dolazi sa zakašnjenjem impuls »prekid programa«. Zakašnjenje u programu je veće od zakašnjenja SP-a, instrukcija ION je neposredno izvršena, te se transfer u subrutinu može dogoditi. Instrukcija IOF (interrupt off) neposredno prije početka mijenjanja sadržaja lokacije N onemogućuje prekid. Sadržaj lokacije N povećava se za jedan i opisani postupak se ponovi. Test $N = 128$ kada je ispunjen znači da je upravo prikazano 128 tačaka s koordinatom $y = \phi$ odnosno sadržajem N_1 jednakim nuli. Tada program briše sadržaj lokacije N i povećava sadržaj lokacije N_1 za jedinicu, a program ispisuje na ekranu katodne cijevi drugu liniju. Nakon prikazane tačke s koordinatama koje odgovaraju sadržaju $N = N_1 = 128$ prikazan je cijeli raster od 128×128 tačaka,

sadržaj lokacije N_1 svodi se na nulu i opisani postupak se ponavlja do prvog prekida. Kao primjer dan je strojni program za generiranje rastera.

```

/
/
/ RASTER 128 x 128
/
RAS, CLA
TAD N
DXL
CLA
TAD N1
DYL
INTR, ION
DIY
NOP
NOP
NOP
NOP
IOF
ISZ N
CLA
TAD RAP
TAD N
SZA
JMP RAS
DCA N
ISZ N1
TAD RAP
TAD N1
SZA
JMP RAS
DCA N1
JMP RAS
N,  $\phi$ 
N1,  $\phi$ 
RAP,  $-2\phi\phi$ 
/
/
/

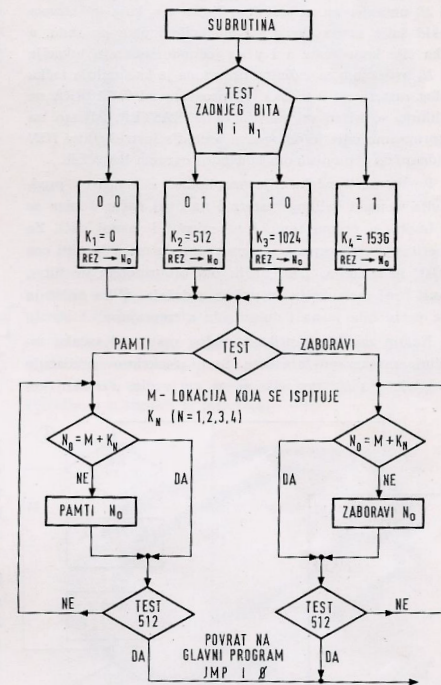
```

2.2. Subrutina

Seдам bitni registar može prikazati decimalni broj 127. Za prikaz svake tačke na ekranu katodne cijevi u rasteru 128×128 potrebna su dva 7 bitna registra za x i y koordinate. Ovakvim načinom pamćenja koordinata x i y jedan deskriptor okupirao bi više od jedne, dakle dvije lokacije memorije, čime bi korisni prostor za sadržaj bio smanjen na polovinu.

Zbog toga subrutina, sl. 4, testira najmanje značajne bitove koordinata x i y i svrstava deskriptore u četiri grupe 00, 01, 10 i 11 ovisno o vrijednosti njihovih zadnjih bitova. Time je apsolutna vrijednost svakog zapamćenog deskriptora određena sadržajem lokacije i grupe u kojoj se ta lokacija nalazi. Uz takav način pamćenja deskriptora u prvoj polovini svake lokacije memorije koja je rezervirana za deskriptore zapisana je 6 bitna x koordinata, a u drugoj 6 bitnoj polovini riječi y koordinata. Prije ispisivanja ovako »krnjeg« deskriptora, odnosno njegovog prikaza na ekranu katodne cijevi dodaju se x i y koordinatama najmanje značajni bitovi ovisno o grupi u kojoj se deskriptor nalazi.

Najmanje značajni bitovi su i jednako vjerojatni, te se grupe bez obzira na oblik slike jednoliko pune. Svaka grupa sadrži 512 lokacija. Njihove relativne adrese označene su na slici. Program za pamćenje pamti svaki deskriptor asocijativno. Program testira prvu lokaciju grupe kojoj deskriptor, smješten privremeno u lokaciju N_0 , pripada. Ako je ta lokacija jednaka nuli, program zapisuje deskriptor u tu lokaciju.



Sl. 4. Potprogram za stvaranje liste

Ako u prvoj lokaciji postoji neki deskriptor program ga uspoređuje s novim deskriptorom. Ako su jednaki novi deskriptor se ignorira, a ako su različiti program sukcesivno ispituje slijedeću lokaciju iste grupe, dok ne dođe do prve čiji sadržaj je jednak nuli. Ako je svih 512 lokacija puno, program ignorira novi deskriptor a lista od 512 postojećih deskriptora ostaje nepromijenjena. Operator to stanje primjećuje po tome što SP na ekranu više ne ostavlja tačkasti trag. Pero je »ostalo bez tintex«.

U programu za brisanje deskriptora sa liste, program stavlja zadnji deskriptor s liste na mjesto upravo izbrisano i osigurava da lista ostane »neprekinuta«. U protivnom moglo bi se dogoditi da kod upisivanja neposredno nakon operacije brisanja budu upisani jednaki deskriptori na različitim mjestima liste. Deskriptor nula ne može biti zapamćen.

Ako se deskriptori očitavaju s perforirane trake program ignorira svaki deskriptor koji ima x ili y koordinatu veću od 127.

2.3. Potprogrami 9 tačaka i lista

Na sl. 5 prikazani su dijagrami toka potprograma 9 TACAKA i LISTA.

Nakon transfera u subrutinu u lokaciji N i N_1 nalaze se koordinate tačke koja je izazvala prekid. Potprogram 9 TACAKA generira pravokutni vijenac od 8 tačica oko centralne tačke koja je izazvala prekid. Prvo se generira tačka čije koordinate x i y su jednake sadržaju lokacije N i N_1 umanjenoj za jedinicu. Tačka $x_0 y_0$ koja je izazvala prekid šalje se na ekran katodne cijevi peta po redu, a tačka čije koordinate x i y su jednake sadržaju lokacije N i N_1 uvećanom za jedinicu prikazana je kao zadnja tačka malog rastera od 9 tačaka. Potprogram od 9 TACAKA ne razlikuje se bitno od potprograma RASTER. Mjesto na potprogramu gdje je omogućen prekid s instrukcijom ION je identično već opisano kod potprograma RASTER.

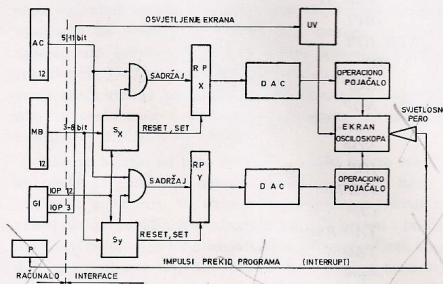
Raster od 9 tačaka nije neophodan. U stvari on predstavlja element velikog rastera i na ovaj način dobiva se na brzini, a prema tome i na mirnoj i jasnoj slici. Za generiranje cjelokupnog rastera računalo mora izvršiti cca 500.000 instrukcija. Tom broju bio bi pridodan ne bitno manji broj instrukcija za prikaz sadržaja. Slika sadržaja više ne bi bila jasna i dolazila bi s treptajima.

Nakon završenog prikaza malog rastera 9 tačaka započinje prikaz sadržaja liste. Svaki deskriptor prebacuje se u lokaciju N_0 , rastavlja se na x i y dio, a svakoj ko-

ordinati x i y pridodaje se vrijednost zadnjeg bita ovisno o grupi kojoj pripada. Sadržaj memorije se pri tom ne mijenja. Potprogram za ispisivanje liste testira sadržaj svake lokacije. Ako je sadržaj nula, prelazi na testiranje slijedeće lokacije. Nakon 2048 prolaza u petlji ispisivanja liste potprogram se vraća na peneriranje malog rastera od 9 tačaka ovisno o rezultatu TESTA 2.

3. Blok shema vezne logike

Na sl. 6 prikazan je pojednostavljeni blok vezne logike (»interface«).

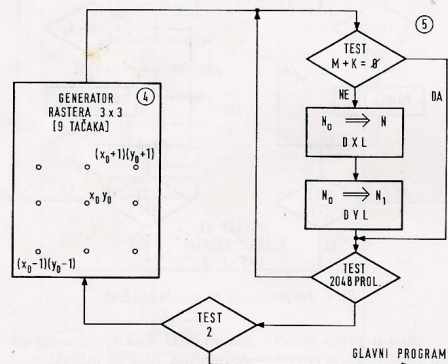


Sl. 6. Blok shema vezne logike (»interface«)

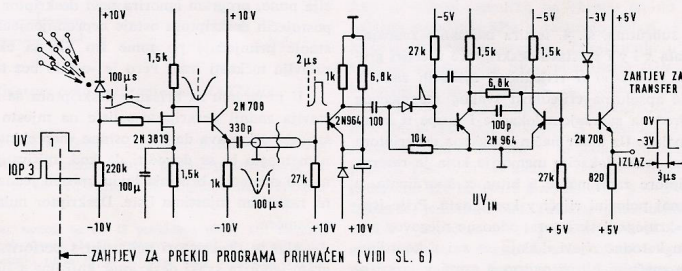
U selektor x i selektor y dolaze impulsi iz memorijskog »buffer« registra MB i generatora impulsa. Prije izvršenja instrukcija DXL i DYL u akumulatoru računala se nalazi koordinata x odnosno y od deskriptora čija će analogna vrijednost izvršenjem instrukcije DIY biti vidljiva na ekranu katodne cijevi (sl. 3).

Izvršenjem instrukcije DXL odnosno DYL u selektor x i selektor y dolaze impulsi $IOP1$ i $IOP2$ pomaknuti međusobno za dio mikrosekunde. Ako se izvršava instrukcija DXL , $IOP1$ preko selektora x resetira RP_x , a impuls $IOP2$ također preko selektora x omogući transfer sadržaja akumulatora na RP_x .

Na isti način izvršenjem instrukcije DYL sadržaj akumulatora prebacuje se na RP_y . Izvršavanjem instrukcije DIY iz generatora impulsa dolazi impuls $IOP3$ koji se preko oblikovača UV šalje na katodu katodne cijevi kao impuls za osvjjetljenje. Sadržaj na registrima za podatke ostaje nepromijenjen do slijedećih instrukcija DXL i DYL .



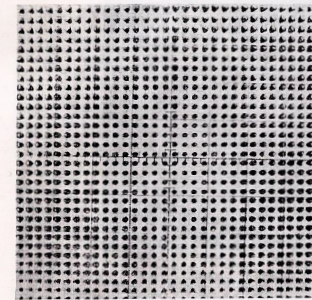
Sl. 5. Potprogram »9 TACAKA« i »LISTA«



Sl. 7. Elektronička shema svjetlosnog pera

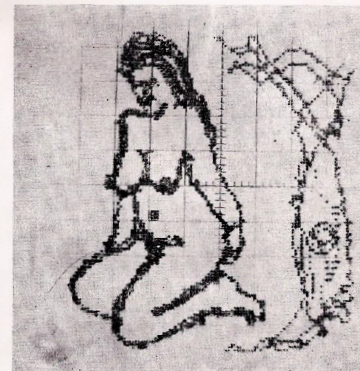
Osciloskop koji se koristi za prikaz sadržaja je TEKTRONIX 536. Između digitalno analognih konvertera i osciloskopa nalaze se operaciona pojačala koja služe kao sumator struja iz digitalno analognih konvertera s izrazito malom ulaznom impedancijom. (Jako negativna povratna veza.)

Kod gradnje SP moguće je svjetlosnim vodom dovesti svjetlo s ekrana na fotomultiplikator i tako izvršiti pretvorbu svjetlosnog signala u električni. Izbor ovog puta povezan je s brzinom, jer se odziv fotomultiplikatora mjeri u dijelovima mikrosekunde. Drugi put je upotreba jeftine fotodiode ili foto-tranzistora koji su do sada bili znatno sporiji.



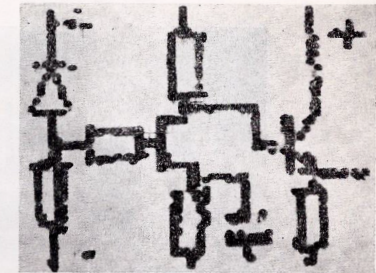
Sl. 8. a) Raster na ekranu katodne cijevi

Rješenje SP (elektronička shema) prikazana je na sl. 7. Korištena je foto-dioda čiji je odziv oko 100 mikrosekundi. Uzastopnim pojačanjem signala sa spore foto-diode i dovođenjem pojačala u zasićenje dobiven je signal čije je vrijeme porasta 50 puta brže. Taj signal okida univibrator UV_{IN} , a impuls s univibratora izvrši prekid. Impuls za osvjjetljenje uzrokovan je impulsom $IOP3$ i traje 5 mikrosekundi. 2–3 mikrosekunde nakon prvog brida impulsa za osvjjetljenje UV , nastaje impuls s univibratora UV_{IN} . To je kašnjenje svjetlosnog pera i u potprogramu RASTER i 9 TACAKA ne smije kroz to vrijeme doći do promjene sadržaja N i N_1 jer bi inače bila zapamćena kriva koordi-



b) Figura nacrtana pomoću svjetlosnog pera

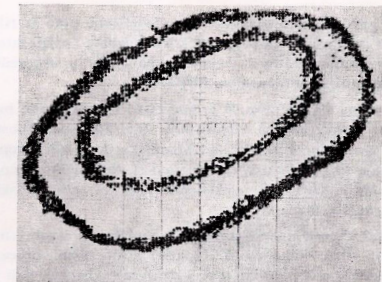
nata odnosno neki drugi deskriptor. Zakašnjenje u potprogramima iznosi 4 mikrosekunde zbog pouzdanosti u radu.



Sl. 8. c) Električka shema svjetlosnog pera, nacrtana na ekranu katodne cijevi

Veoma brzi novi FOTO-FET-ovi (Photo field effect transistor) čiji je odziv u nanosekundnom području ubrzali bi odziv i u potprogramima bi moglo biti izbačeno zakašnjenje. Cijeli sistem vidi se na sl. 8 e.

U lijevom dijelu slike je računalo PDP-8, u desnom dijelu slike osciloskop na čijem ekranu je svjetlosnim perom (nalazi se u ruci korisnika) nacrtana kvalitativna krivulja, a u sredini je interface.



S. 8. d) Simulacija kompleksnih krivulja gustoće vjerovatnosti

4. Zaključak

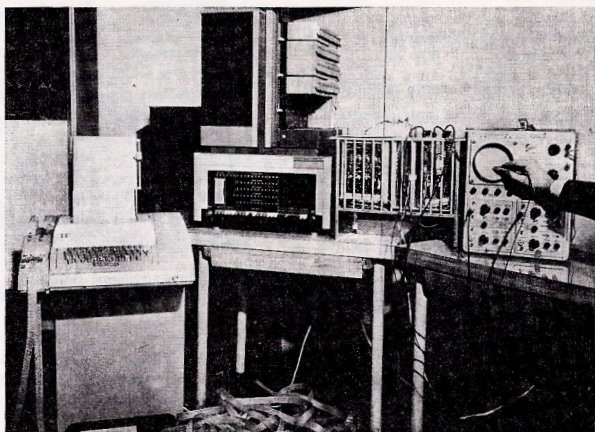
Grafičko unošenje podataka u računalo daje niz novih mogućnosti korisniku. Korisnik pomoću svjetlosnog pera može izdvojiti interesantne dijelove slike (krivulja, dijagram, spektar) i tako na najprikladniji način pratiti promjenu procesa. I preko najbrže periferne jedinice spojene na računalo komunikacija čovjek – računalo odvijala bi se znatno sporije. Kod grafičke komunikacije čovjek i računalo »govore« istim jezikom i ukoliko je proces grafički prikladno prikazan korisnik ima u svakom času pregled i kontrolu nad cijelim procesom (direktna kontrola – eksperimena, drugih matematskih operacija industrijske proizvodnje, reagiranja drugog čovjeka ili grupe ljudi itd.).

U izvedenom i opisanom sistemu riješen je niz tehničkih problema kako u softveru tako i u hardwareu.

S potprogramom 9 TACAKA dobijeno je na brzini čije prednosti su već opisane u članku. Povećan je kapacitet memorije memoriranjem »krnjih« deskriptora. Osiguran je rad prekida koji može nastati samo na određenom mjestu

način gradnje bio bi veoma skup i dugotrajan. Zbog toga se izvori podataka simuliraju.

Na sl. 8 d prikazan je kompleksni spektar koji simulira podatke iz nuklearnog stroja reprezentirajući amplitudnu



Sl. 8. e) Izgrađeni sistem za optičku komunikaciju s računalom. Od desna na lijevo: ekran osciloskopa i crtanje sa svjetlosnim perom; »interface«; laboratorijsko računalo PDP-8; teleprinter računala

pa su vanjske smetnje isključene. Svjetlosno pero je riješeno s malim brojem poluvodičkih elemenata i cijeli interface je veoma jeftin i zanemariv po cijeni u usporedbi s Tektronixom ili računalom PDP-8.

Ovakav sistem moguće je koristiti i u medicinsko-biološkim istraživanjima. Ispitivanje ljudskih sposobnosti, mjerenje brzine reakcije kod kompleksnih radnji, prepoznavanje sličnih detalja, asocijacije i sl. Nadalje korištenje prilikom konstrukcija električnih krugova, građevinskih konstrukcija, mapa. Sl. 8 c.

Kod skupljanja podataka iz kompleksnog izvora (nuklearni stroj, industrijski, biološki i društveni proces) neophodno je obradu vršiti preko mjernog sistema, koji je teško izgraditi bez uključenog izvora podataka. Takav

distribuciju. Uz takvu simulaciju u Institutu »Ruder Bošković« u završnoj fazi izgradnje je sistem za megakanalnu rezoluciju i randomizaciju spektra koja će rezultirati u jednolikom odnosno optimalnom korištenju memorije računala.

LITERATURA

1. B. Souček, K. Culjat, V. Bonačić: Digitalna računala u mjerenju i regulaciji, isti broj ovog časopisa.
2. K. Culjat, V. Bonačić, B. Souček: Redukcija i sortiranje podataka eksperimenta računalom, isti broj ovog časopisa.

NASLOV AUTORA:

V. Bonačić — B. Souček — K. Culjat, Institut »Ruder Bošković«, Zagreb