

## NOVI SOFTVERSKI SISTEM ZA PODRŠKU PROCESA FUNKCIONALNE ELEKTRIČNE TERAPIJE

Strahinja Došen, *Fakultet tehničkih nauka, Novi Sad*

Dejan B. Popović, *SMI, Dept. of Science and Technology in Health, Aalborg, Denmark*

Nikola Jorgovanović, *Fakultet tehničkih nauka, Novi Sad*

**Sadržaj** – U ovom radu prikazan je softverski sistem za računarsku podršku procesu Funkcionalne Električne Terapije (FET). FETStudio implementira nov koncept grafičkog podešavanja protokola stimulacije, kao i sistematsku podršku drugim fazama procesa rehabilitacije. Sistem obezbeđuje manipulaciju multimedijalnim podacima koji se koriste u okviru FETa, kao i vođenje evidencije o kompletnoj istoriji tretmana. Sistem omogućava programiranje široke klase stimulatora i kao takav je potencijalno interesantan ne samo lekarima, već i projektantima NP.

### 1. UVOD

Funkcionalna električna terapija (FET) predstavlja jednu od terapijskih procedura u neurorehabilitaciji. Neurorehabilitacija se odnosi na metode i tehnologije koje imaju za cilj poboljšanje performansi očuvanih neuromuskularnih struktura kod ljudi koji su pretrpeli povredu ili oboljenje centralnog nervnog sistema. FET uključuje kombinaciju funkcionalne električne stimulacije koja se primenjuje za izazivanje prirodnih pokreta subjekta i intenzivnih vežbi [1]. Neuroproteza (NP) je uređaj za generisanje električne stimulacije i osnovni je terapijski instrument u okviru FETa. Personalni računar (PC) je često prisutan kao podrška korisniku pri upotrebi NP i primeni terapije [2]. Predmet ovog rada je upravo deo sistema koji se nalazi na PC računaru. U nastavku dajemo pregled nekih novih softverskih rešenja koja prate sisteme za FET, i koja se nalaze u praktičnoj kliničkoj primeni.

### 2. POSTOJEĆA REŠENJA

*BionicGlove* je trokanalni elektronski stimulator za kontrolu otvaranja i zatvaranja šake. Ugao fleksije i ekstenzije ručnog zglavka se pretvara u voljno kontrolisani diskretni komandni signal za aktiviranje stimulacije koja ispravlja ili savija prste i palac. Softver koji se isporučuje uz ovaj stimulator je napisan u jeziku *Visual Basic* i nudi jednostavan i skroman korisnički interfejs. Stanje senzora prikazuje se grafički u obliku vremenski promenljivog signala. Parametri stimulacije zadaju se unošenjem numeričkih vrednosti, preko tri reda numeričkih kontrola. Za svaki kanal može se definisati intenzitet stimulacije zadavanjem amplitude ili širine impulsa. Tako definisan stimulatorni protokol moguće je poslati na uređaj, preko jednog od serijskih portova računara [3].

Na Elektrotehničkom fakultetu u Beogradu je razvijen elektronski stimulator nazvan *Mrdalica* sa 16 galvanski odvojenih izlaznih kanala i 14 ulaznih – 6 digitalnih i 8 analognih. Za realizaciju protokola stimulacije koristi se i pet tajmera, kao i pet digitalnih kontrolnih signala [4]. Aplikacija koja se dobija uz uređaj, napisana je u jeziku *Turbo Pascal* i izvršava se pod DOS operativnim sistemom. Program nudi

nekoliko ekrana putem kojih se definišu parametri rada uređaja. Prvo se definiše konfiguracija uređaja – izaberu se kanali, tajmeri i kontrolni signali koji će biti korišćeni. Zatim se prelazi u prozor za nameštanje pragova na analognim ulaznim kanalima. Pragovi se unose u vidu kvantizacionih nivoa A/D konvertora. Protokol stimulacije zadaje se tako što se aktivnost stimulatora opisuje konačnim automatom stanja. Prvo se zadaju uslovi ulaska u dato stanje – stanja digitalnih i analognih ulaza, vrednosti tajmera i stanje kontrolnih signala. Zatim se zadaje intenzitet i frekvencija stimulacije na svakom od izlaznih kanala. Ovako definisan protokol stimulacije može se snimiti u fajl ili poslati na uređaj putem nekog od serijskih portova.

*Complex Motion* je višenamenski programabilni transkutani električni stimulator. Uređaj ima četiri izlazna i dva ulazna kanala. Protokol stimulacije se zapisuje na inteligentnu čip karticu koja se postavlja u slot uređaja. Softver koji prati NP je realizovan u programskom okruženju *LabView*. Stimulatorni protokol se definiše sekvencijalnim slaganjem primitiva (primitivi su prikazani u obliku ikona) za svaki kanal stimulacije nezavisno. Primitivi definišu promenu stimulacije na kanalima, ili neku drugu od korisničkih ili vremenskih sekvenci. Svaki primitiv je karakterisan skupom parametara [5].

Mikrominijaturni električni stimulator *BION*, spada u grupu implantabilnih sistema, koji su takođe podržani računarnom. Napajanje implanata vrši eksterni modul, koji može da upravlja sa najviše 256 *BION* jedinica. Implantacija, postavljanje i monitoring *BION* sistema se obavlja uz pomoć *ClinFit* softvera na prenosnom PC računaru. *ClinFit* je Windows aplikacija, napisana u Visual Basic-u. Program koristi *Access* relacionu bazu podataka kako bi kreirao i održavao zapise o mestima implantacije, testovima praga nadražaja, programima vežbe koji su prepisani pacijentima, kao i zapažanja o praktičnom korišćenju sistema od strane pacijenata [6].

### 3. ANALIZA POSTOJEĆIH REŠENJA

Sva predstavljena softverska rešenja nude praktično samo jednu osnovnu funkciju – programiranje NP. U nekim rešenjima postoji podrška za formiranje i održavanje baze pacijenata i/ili stimulatornih protokola, ali takođe u veoma ograničenom obliku – implementirane su osnovne funkcije i vrlo jednostavne strukture podataka.

U rešenjima *BionicGlove* i *Mrdalica* stimulatorni protokol zadaje se numerički, unosom odgovarajućih parametara. Pri tome se u programu *Mrdalica* veličine definišu i zadaju u merama koje zahtevaju određeno tehničko predznanje, kao i poznavanje unutrašnje strukture i realizacije stimulatora (npr. prag A/D konverzije, stanje tajmera i kontrolne linije). *Complex Motion* omogućava zadavanje protokola putem skupa primitiva. Kada se uzme u obzir da ima i do 56 različitih primitiva, jasno je da zadavanje

protokola na ovaj način zahteva određenu veštinu i snalažljivost. Takođe, ni tumačenje protokola koji je ovako opisan nije trivijalna procedura. Iako *ClinFit* i *CompexMotion* softver omogućavaju grafičku vizualizaciju protokola, ni jedno rešenje ne podržava grafički način njegovog definisanja.

Zajednička karakteristika svih rešenja je da podržavaju programiranje samo jednog konkretnog tipa stimulatora, uz koji se i isporučuju.

#### 4. NOVA KONCEPCIJA

Na osnovu analize postojećih rešenja, postavljeni su zahtevi koje treba da ispuni novi sistem za softversku podršku NP. Mogućnosti savremenih računara i okruženja za razvoj softvera su takve da je podržan razvoj grafički orijentisanih aplikacija, koje upravljaju multimedijalnim podacima.

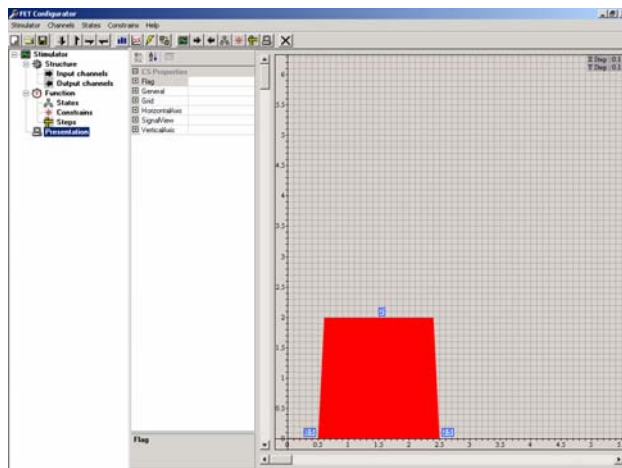
Osnovni zadatak koji je postavljen novom sistemu je razvoj novog, neposrednog i intuitivnog, korisničkog interfejsa za definisanje protokola stimulacije, koji će biti lako razumljiv i prihvatljiv lecarskom osoblju. Razvijen je koncept grafičkog interfejsa koji omogućava podešavanje aktivnosti kanala direktnim intervencijama na grafičkoj prezentaciji.

Novi softverski sistem ne samo da omogućava definisanje protokola i prenos parametara sa računara na NP, već i sistematsku podršku svim drugim fazama FETa: prijem pacijenta, formiranje elektronskog kartona, održavanje baze kartona i protokola, procenu stanja pacijenta, pamćenje istorije tretmana. Pri tome se u oblasti FET kao česta sredstva ilustracije stanja pacijenta koriste video snimci, crteži, fotografije. Zbog toga je posebna pažnja u razvoju softvera poklonjena obezbeđenju rada sa multimedijalnim podacima. U svrhu egzaktne numeričke ocene stanja pacijenta u okviru FET se koristi veliki broj testova. Iako se oni razlikuju po sadržaju pitanja (zadataka), struktura testova je obično vrlo slična. Zato je odlučeno da će novi sistem obezbediti laku integraciju i korišćenje standardnih i specijalizovanih korisnički kreiranih testova.

Na Katedri za automatiku i upravljanje sistemima, Fakulteta tehničkih nauka Univerziteta u Novom Sadu, razvijen je jedan programabilni stimulator. Ovaj originalni FET stimulator [10], i njegova klinička primena su bili neposredna inspiracija za ovaj istraživački projekat. Iako postoji velika raznovrsnost na polju hardverske realizacije FET stimulatora, ponašanje značajnog broja uređaja može se opisati istim modelom – konačnim automatom stanja [1]. Signali na ulaznim kanalima stimulatora izazivaju prelazak NP u određeno stanje u kome se aktivira predefinisana šema stimulacije na njegovim izlaznim kanalima. Iz tog razloga, novi softverski sistem nije vezan za jedan konkretan tip stimulatora, već omogućava programiranje široke klase stimulatora čije se ponašanje može opisati istim modelom.

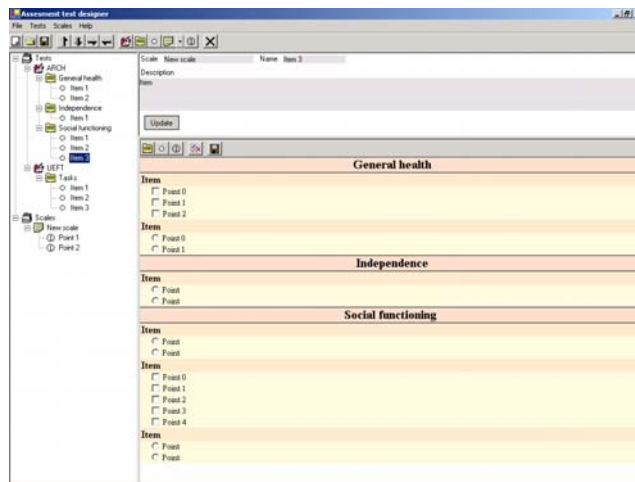
#### 4. REALIZACIJA SISTEMA

Kako bi se implementirali postavljeni zahtevi, softverski sistem *FETStudio* je realizovan u obliku tri nezavisna softverska modula: *FETConfigurator*, *FETAssesmentTestDesigner* i *FETStudioFrontEnd*.



Sl 1. Glavni prozor aplikacije *FETConfigurator*

Prve dve aplikacije koristi projektant NP. Modul *FETConfigurator* (slika 1) omogućava korisniku da opiše statički i dinamički model konkretne NP. Projektant definiše ulazne i izlazne kanale, stanja u kojima se može naći NP, korake promene vremenskih i električnih veličina koje podržava uređaj, podrazumevanu stimulaciju na svakom kanalu u svakom stanju, ograničenja vremenskih i električnih veličina na pojedinim kanalima i u pojedinim stanjima, kao i grafički izgled i osobine kontrole za definisanje stimulacije na jednom kanalu stimulatora. Sve ove informacije zapisuju se u specijalni konfiguracioni fajl.



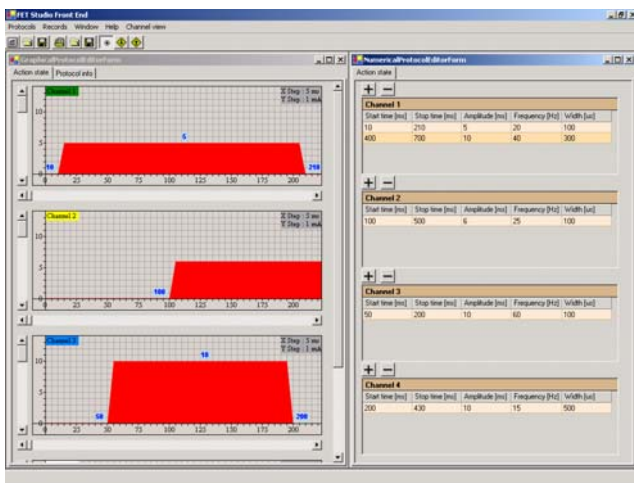
Sl 2. Glavni prozor aplikacije *FETAssesmentTestDesigner*

Modul *FETAssesmentTestDesigner*, prikazan na slici 2, omogućava projektantu NP da definiše strukturu, sadržaj i bodovanje testova za procenu stanja pacijenta, koji će biti korišćeni uz njegovu neuroprotezu. Projektant organizuje test pitanja (zadatke) u različite grupe i zadaje skalu za ocenu svakog pitanja (zadatka). Definisani testovi zapisuju se u konfiguracioni fajl.

Aplikacija *FETStudioFrontEnd* se, zajedno sa konkretnim stimulatorom, isporučuje krajnjem korisniku tj. lekaru. Uz softver se dodaju i dva konfiguraciona fajla koja su izlaz prethodnih programa. Aplikacija *FETStudioFrontEnd* prilikom startovanja učitava informacije iz konfiguracionih fajlova: model stimulatora i definisane testove.

Na osnovu tih informacija modul prilagođava izgled interfejsa za definisanje protokola stimulacije potrebama

konkretne NP uz koju je i isporučen. Aplikacija omogućava korisniku i grafički i numerički način zadavanja parametara protokola, uz poštovanje svih ograničenja koje nameće sam uređaj i njegova predviđena klinička primena. U grafičkom editoru protokola, aktivnost jednog kanala u nekom stanju prikazuje se u posebno razvijenoj grafičkoj kontroli. Stimulacijska aktivnost prikazana je kao signal sastavljen od jednog ili više pravougaonika različite amplitude. Svaki pravougaonik predstavlja jedan niz stimulacionih impulsa. Lekar zadaje stimulaciju pomeranjem ivica pravougaonih delova signala, čitavih pravougaonika, kao i dodavanjem i brisanjem perioda stimulacije. Editor omogućava prikazivanje stimulacione aktivnosti drugih kanala, kao statičnih slika, u okviru kontrole za dati kanal, kako bi se aktivnost datog kanala mogla referencirati u odnosu na druge kanale. Aplikacija omogućava dvosmernu komunikaciju sa uređajem. Korisnik može da održava bazu standardnih i/ili često korišćenih stimulacionih protokola. Na slici 3 je prikazan glavni prozor aplikacije, sa grafičkim i numeričkim editorom protokola.



Sl 3. Glavni prozor aplikacije FETStudioFrontEnd

U okviru modula implementiran je i elektronski karton pacijenta. Pored osnovnih ličnih i dijagnostičkih informacija, karton odebduje praćenje kompletne istorije tretmana. Za svaki pregled pacijenta pored osnovnih podataka o pregledu, mogu se vezati multimedijalni sadržaji koji dokumentuju stanje pacijenta – slika, video film ili test.

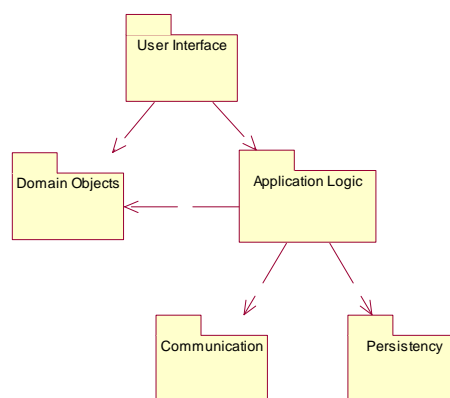
## 6. PRIMENJENE SOFTVERSEKE TEHNOLOGIJE

Softverski sistem je realizovan korišćenjem Visual Studio .NET razvojnog okruženja, u programskom jeziku C#. Za modelovanje sistema korišćen je UML. U projektovanju i razvoju sistema primenjeni su principi USDP (*Unified Software Development Process*) [7]. Sistem ima komponentnu arhitekturu, iterativno i inkrementalno je razvijan, pri čemu su u svakom podsistemu prvo razvijeni elementi kritične funkcionalnosti. Arhitektura sistema, osnovni mehanizmi interakcije i struktura pojedinih delova sistema, osmišljeni su uz korišćenje nekih standardnih strukturalnih, bihevioralnih i arhitekturnih šablona [8]. Za raspodelu funkcija po klasama sistema korišćeni su šabloni za raspodelu odgovornosti [9].

## 5. UNUTRAŠNJA STRUKTURA SISTEMA

Pojedinačne aplikacije su sastavljene od podsistema od kojih su neki zajednički za sve module. Analizom FET domena i modelovanjem njegove strukture i procesa, uočeni su koncepti od interesa. Ti koncepti su realizovani softverskim klasama koje čine podsistem domena. Osnovne grafičke i numeričke kontrole koji su delovi numeričkog i grafičkog editora protokola stimulacije, čine podsistem pogleda na objekte domena. I jedan i drugi podsistem koriste svi moduli FETStudio sistema.

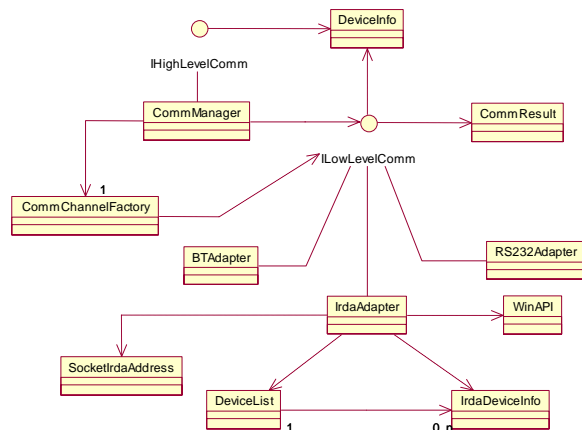
Unutrašnja struktura modula FETStudioFrontEnd je sačinjena od pet podsistema: korisnički interfejs, domen, aplikativna logika, komunikacija, perzistencija. Ovi podsistemi organizovani su u skladu sa arhitekturnim šablonom – raspodela sistema u funkcionalno odvojene slojeve [7]. Na slici 4 prikazani su pomenuti podsistemi povezani relacijama zavisnosti.



Sl 4. FETStudio podsistemi

Klase korisničkog interfejsa i domena organizovane su po MVC (*Model View Controller*) šablonu. Primena ovog šablona obezbeđuje postojanje i automatsku sinhronizaciju više pogleda na isti stimulacioni protokol. Podsistem aplikativne logike sastavljen je od klasa kontrolera (*kontrolerski* šablon raspodele odgovornosti). Objekti kontroleri primaju informacije o korisničkim akcijama od objekata korisničkog interfejsa i koristeći objekte nižih slojeva, obrađuju zahteve korisnika. Na ovaj način vrši se rasporezanje korisničkog interfejsa i aplikativne logike. Kako bi se obezbedila podrška što široj klasi NP, sistem je projektovan sa namerom da se obezbedi korišćenje različitih komunikacionih interfejsa. Pristup podsistemu se vrši preko jednog ulaznog *fasadnog* objekta koji nudi interfejs visokog nivoa apstrakcije. Detalji konkretnih komunikacionih linkova enkapsulirani su u posebne klase koje implementiraju zajednički interfejs niskog nivoa. Fasadni objekat implementira takođe i *Most* šablon, kako bi ostatak sistema bio neosetljiv na promenu komunikacionog interfejsa. Pristup pojedinim Windows API funkcijama, koje su neophodne kako bi se realizovala komunikacija infracrvenom vezom, realizovan je posebnim *adapterskim* klasama. Dijagram klase komunikacionog podsistema prikazan je na slici 5. Podsistem perzistencije organizovan je slično kao i podsistem komunikacije, kako bi se omogućilo korišćenje različitih fizičkih mehanizama pamćenja protokola i kartona pacijenata. Fasadni objekat koji je ulaz u podsistem, koristi isključivo apstraktne .NET interfejsa za pristup provajderima podataka. Specijalna fabrička klasa, na zahtev fasadnog

objekta, kreira primerke konkretnog provajdera podataka. Struktura svih podsistema sa dijagramima klasa data je u [11].



Sl 5. Komunikacioni podsistem

## 5. ZAKLJUČAK

U radu je prikazan novi softverski sistem za podršku procesa funkcionalne električne terapije. *FETStudio* je programska podrška koja je otklonila niz nedostataka i problema uočenih analizom postojećih rešenja primenjenih u praksi. Program se oslanja na mogućnosti savremenih računarskih sistema u realizaciji kvalitetne grafike i manipulaciji multimedijalnim podacima. *FETStudio* implementira nov koncept grafičkog načina definisanja stimacionog protokola, a podržava i ostale faze FET procesa. Na taj način, sistem predstavlja računarsku podršku medicinskom osoblju. Sistem je koncipiran na način koji omogućuje primenu na celoj klasi elektronskih stimulatora, i što je još važnije seriji postojećih i budućih kliničkih protokola. Kako su svi moduli razvijeni pod .NET platformom, prirodna evolucija sistema je idealna za dalje primene u telemedicini i telekonsultacijama (npr. razmena protokola stimulacije, podešavanje protokola i uređaja sa udaljene lokacije, kao i prenos video snimka efekta stimulacije nazad na udaljenu lokaciju).

## 5. LITERATURA

- [1] D. Popović, T. Sinkjær, "Control of Movement for the Physically Disabled", Springer-Verlag, London, 2000 godine, ISBN 1852332794
- [2] Y. Handa at al., "Functional Electrical Stimulation Systems for Restoration of Motor Function of Paralyzed Muscles – Versatile Systems and a Portable System", Frontiers Med Biol Engng. 4:241-255, 1992

- [3] A. Prochazka, M. Gauthier, M. Wieler, Z. Kenwell, "The Bionic Glove: an Electrical Stimulator Garment that Provides Controlled Grasp and Hand Opening in Quadriplegia", Arch Phys Med Rehabil. 78:608 – 614, 1997
- [4] N. Fišeković, D. Popović, "New Controller for Functional Electrical Stimulation Systems", Med Eng Phys. 23:391-399, 1999.
- [5] T. Keller, M.R. Popović, I.P.I. Pappas, P.Y. Muller, "Transcutaneous Functional Electrical Stimulator Compex Motion", Art Organs. 26:219-223, 2002
- [6] G.E. Loeb, R.A. Peck, W. H. Moore, K. Hood, "BION System for Distributed Neural Prosthetic Interfaces", Med Eng Phys, 23:9-18, 2001.
- [7] I. Jacobson, G. Booch, J. Rumbaugh, "The Unified Software Development Process", Addison-Wesley, 1999, ISBN 0-201-57169-2
- [8] E. Gamma, R. Helm, R. Johnson, J. Vlissides, "Design Patterns – Elements of Reusable Object Oriented Software", Addison-Wesley, 1995, ISBN 0-201-63361-2
- [9] C. Larman, "Applying UML and Patterns", Prentice Hall, 2002, ISBN 0-13-092569-1
- [10] N. Jorgovanović, "Upravljanje Funkcionalnom Električnom Stimulacijom za Neurorehabilitaciju Pokreta", Doktorska disertacija, Univerzitet u Novom Sadu, 2003.
- [11] S. Došen, "Softverski sistem za podršku procesa funkcionalne električne terapije", Magistarski rad, Univerzitet u Novom Sadu, 2004.

**Zahvalnica:** Rad na ovom projektu su delimično pomogli Ministarstvo za nauku i zaštitu životne sredine Srbije, Beograd, i Danish National Reseach Foundation, Danska.

**Abstract** – We present a new software termed FETStudio for personal computer support of Functional Electrical Therapy (FET). FETStudio implements a new concept for the graphical adjustment of stimulation protocol, as well as systematic support for other phases of the FET. The tests of the applicability were performed with the FET systems [10]. FETStudio can manipulate multimedia data of interest for FET and IT in health. FETStudio allows the programming of a whole class of stimulator models, as well as, various clinical procedures with these stimulators. These features make the FETStudio a valuable tool for the rehabilitation of humans with sensory-motor disabilities.

## A NOVEL SOFTWARE SYSTEM FOR SUPPORT OF FUNCTIONAL ELECTRICAL THERAPY PROCESS

Strahinja Došen, Dejan B. Popović,  
Nikola Jorgovanović