

Robotski krmilnik z odprto arhitekturo vodenja zasnovan za industrijski manipulator Motoman MH5

Peter Čepon, Matjaž Mihelj, Marko Munih

Univerza v Ljubljani, Fakulteta za elektrotehniko, Tržaška 25, 1000 Ljubljana

peter.cepon@robo.fe.uni-lj.si

Open Architecture xPC Target Based Robot Controllers for Industrial and Research Manipulators

Industrial robot manipulators have reached a stage where their mechanical characteristics are well optimized. However, industrial hardware and software control architectures are rarely open for sensor integration. In order to overcome these constraints, custom-designed hardware interfaces and software algorithms need to be designed. The paper summarizes the implementation of open architecture xPC Target based robot controllers for industrial and research manipulators. Custom hardware solutions are presented.

Kratek pregled prispevka

V današnjem času so industrijski roboti dosegli stopnjo, kjer so njihove mehanske karakteristike zelo dobro optimizirane. Medtem, ko so proizvajalci dobro optimizirali mehanske karakteristike, pa so na drugi strani ves čas ohranjali zaprto arhitekturo krmilnikov in s tem skoraj enake funkcionalnosti, kot pri prvih industrijskih robotih. Če hočemo industrijske robote uporabljati v raziskovalne namene, nam tako ne preostane drugega kot, da se lotimo z načrtovanjem strojnih modulov in razvojem programske opreme po meri. V tem članku je opisan razvoj in izbira strojne opreme za izgradnjo xPC Target krmilnika z odprto arhitekturo in princip delovanja varnostnega mehanizma. Vsa strojna oprema je bila zgrajena tako, da je delovala na xPC target operacijskem sistemu.

1 Uvod

V današnjem času so industrijski roboti dosegli stopnjo, kjer so njihove mehanske karakteristike zelo dobro optimizirane. Medtem, ko so proizvajalci dobro optimizirali mehanske karakteristike, pa so na drugi strani ves čas ohranjali zaprto arhitekturo krmilnikov. Arhitektura krmilnikov industrijskih robotov nam zato, kljub neprestanemu napredku, ponuja skoraj enake funkcionalnosti, kot pri prvih industrijskih robotih. Te funkcionalnosti industrijskih robotov so tako še vedno omejene predvsem na operacije, kot so sledenje trajektorijam, pozicijska kontrola, itd.. Zaradi različnih vzrokov ohranjajo proizvajalci arhitekturo zaprto in jo zelo redko odprejo za integracijo dodatnih senzorjev [1]. To pa pomeni, da je razvoj in implementacija različnih načinov vodenja z uporabo dodanih senzorjev zelo omejena in v nekaterih primerih skoraj neizvedljiva.

Večina večjih proizvajalcev industrijskih krmilnikov se običajno odloča za zaprto arhitekturo krmilnikov. To pomeni, da uporabniki nimajo zunanjega dostopa do krmilnih signalov, s katerimi bi lahko nastavljali parametre manipulatorja. Vendar pa se med proizvajalci najdejo tudi takšni, ki imajo v svoji ponudbi robotske krmilnike z odprto arhitekturo. Na žalost se izkaže, da so pri odprtih krmilnikih frekvence osveževanja majhne ter omejujejo samo izvrševanje zahtevnejših nalog.

Proizvajalca, ki v svoji ponudbi ponujata industrijske robotske krmilnike z odprto arhitekturo sta Stäubli (Stäubli Faverges SCA) in Yaskawa (Yaskawa Electric Corporation). Prvi ponuja družino krmilnikov CS8, ki imajo odprti dostop in so namenjeni predvsem razvoju in raziskovalni sferi. Ti imajo implementiran nivo za vodenje robota v realnem času (*ang. Real-time robot control abstraction layer*) in nižje nivojski vmesnik (*ang. Low-level interface - LLI*), ki omogoča zunanje nastavljanje referenc pozicij in hitrosti sklepov robota [2]. Pri tem je frekvenca osveževanja takšnega krmilnika omejena na približno 1kHz. Pri Yaskawi je

uporabniku na voljo podoben visoko hitrostni modul za sinhroni nadzor programske opreme (*ang. High-speed robot synchronous control software*) [3], ki je sestavljen iz deljenega spominskega vmesnika med robotskim krmilnikom in glavnim računalnikom. Ta shranjuje vse informacije sinhronnega vodenja procesne enote Yaskawa robotskega krmilnika. Glavni računalnik si vse informacije z Yaskawa robotskim krmilnikom izmenjuje preko Yaskawa visoko hitrostnega sinhronnega računalnika (*ang. high-speed synchronous computer - HSC*). Na računalniku lahko teče operacijski sistem v realnem času ali nerealnem času, vendar pa je potrebno povedati, da je 2 ms sinhronizacija s HSC absolutno zagotovljena le v primeru, ko teče operacijski sistem v realnem času. Glavno omejitev HSC krmilnika predstavlja tako povezava med njim in glavnim računalnikom, saj je njena frekvenca osveževanja le 500 Hz.

Zaradi nenehnega razvoja in omejenosti uporabe industrijskih robotskih krmilnikov se velikokrat izkaže, da tudi rešitve, ki jih ponudijo proizvajalci robotskih krmilnikov niso zadostne. Raziskovalci se za potrebe raziskav odločijo za zamenjavo ali nadgradnjo robotskih krmilnikov z obstoječo arhitekturo. Običajno si razvijalci načrtajo celotno strojno opremo z vsemi potrebnimi vmesniki prilagojenimi razvojnemu okolju s programsko opremo, ki so si jo predhodno izbrali za razvoj aplikacij. Takšen princip izgradnje krmilnikov z odprto arhitekturo omogoči razvijalcu dostop in nastavljanje večino uporabnih funkcij manipulatorja ter mu dovoljuje direktno vodenje aktuatorjev v sklepah robota s hitrostmi ali navori. Pri izgradnji krmilnikov po meri je običajno omogočen direkten dostop do servo-ojačevalnikov manipulatorja, kar omogoča razvoj kompleksnejših metod vodenja, integracijo različnih senzorjev (sila/navor, razdalja, vid, zvok) in njihovo ovrednotenje. Takšni robotski krmilniki, ki jih naredimo po meri pridejo še posebej v poštev, ko razvijamo aplikacije, ki dovoljujejo interakcijo med človekom in robotom na podlagi fizičnih kontaktov med njima.

Robotski krmilniki z odprto arhitekturo omogočajo uporabniku nove možnosti razvijanja kompleksnih razvojnih in izobraževalnih aplikacij. Pri vsem tem je potrebno poudariti, da postanejo s povečevanjem kompleksnosti industrijskih aplikacij, omejitve industrijskih robotskih krmilnikov vedno bolj očitne. Takšen problem je predstavljen v članku [4], kjer avtor opisuje aplikacijo merjenja z robotom. Pri tej aplikaciji se izkaže, da kljub zadovoljivi natančnosti robota, zvezno merjenje sivih rešetk z želeno hitrostjo ni mogoče, saj je osveževanje branja pozicij robota prepočasno. Za pravilno delovanje aplikacije je bila zato potrebna modifikacija programske opreme robotskega krmilnika (modifikacija brez podpore s strani proizvajalca), ki je omogočila povečanje vzorčne frekvence pozicije na 500 Hz.



Slika 1: Roboti za katere so bili razviti krmilniki z odprto arhitekturo

Z neprestanim razvojem postajajo tudi roboti vse inteligentnejši, zaradi tega se pričnejo pojavljati zahteve po novih kompleksnejših

aplikacijah. V bodočnosti lahko zato pričakujemo, da bosta človek in robot sodelovala in delovala v istem okolju. V namen izvrševanja uspešnih interakcij med robotom in človekom je zato potreben razvoj novih metod vodenj manipulatorjev.

V Laboratoriju za robotiko na Fakulteti za elektrotehniko se z razvojem robotskih krmilnikov z odprto arhitekturo ukvarjamo že vrsto let. Prvi namenski robotski krmilnik je bil v laboratoriju razvit že v začetku osemdesetih let [5], prvi robotski krmilnik z odprto arhitekturo pa v začetku devetdesetih [6]. V zadnjih 10 letih so bili za namen različnih raziskav, na različnih manipulatorjih razviti robotski krmilniki z odprto arhitekturo. Na začetku so bili krmilniki z odprto arhitekturo osnovani na RT Linux platformi, na kateri je bilo narejenih veliko število raziskav in razvojnih nalog [7, 8, 9, 10]. Zaradi uporabniku neprijaznega programskega okolja za hitro razvijanje, preizkušanje in ovrednotenje sistemov, je bila večina sistemov iz RT Linux platforme prestavljena na xPC Target platformo. S tem se je razvijanje sistemov premaknilo iz programskega okolja C v programsko okolje Matlab, ki je uporabniku prijaznejše za razvijanje in ovrednotenje sistemov.

V nadaljevanju članka je opisan razvoj in izbira strojne opreme za izgradnjo xPC Target krmilnika z odprto arhitekturo in princip delovanja varnostnega mehanizma. Z xPC Target krmilnikom je možno dostopati in nastavljati parametre servo krmilnikov, ki krmilijo dva Motoman-ova robotska manipulatorja MH5. Vse komponente za izgradnjo xPC Target krmilnika po meri in varnostnega modula, ki so bile implementirane pri izgradnji krmilnika, so bile izbrane in razvite za delovanje na xPC Target platformi.

2 Arhitektura strojne opreme

Pri mnogih proizvajalcih je v navadi, da neradi posredujejo tehnične podrobnosti o povezovanju njihovih industrijskih krmilnikov in možnosti uporabe zunanjih vhodnih signalov. Tako je v večini primerov od proizvajalca težko

pridobiti uporabne informacije, ki bi dovoljevale spremembe ali posredovanje v njihovo strojno opremo in bi na takšen način uporabniku olajšali delo pri izgradnji robotskega krmilnika po meri. Problem proizvajalčevega neposredovanja informacij postane še posebej problematično pri novejših industrijskih krmilnikih, ki uporabljajo za komuniciranje z manipulatorji namesto dosedanjih analognih signalov različna digitalna vodila. Ta so v večini primerih patentirana in prirejena za posameznega proizvajalca. Pri tem se je potrebno zavedati, da imajo takšna digitalna vodila velikokrat omejeno osveževanje tako, da je priklop dodatnih naprav na vodilo lahko problematičen in omejuje delovanje letih. Iz stališča razvijalca to pomeni, da je le-ta omejen pri razvijanju zahtevnejših aplikacij, kot je vodenje manipulatorjev z integracijo senzorjev, ki delujejo v realnem času. Vse te pomanjkljivosti imajo velikokrat negativni vpliv pri izgradnji robotskih krmilnikov po meri.

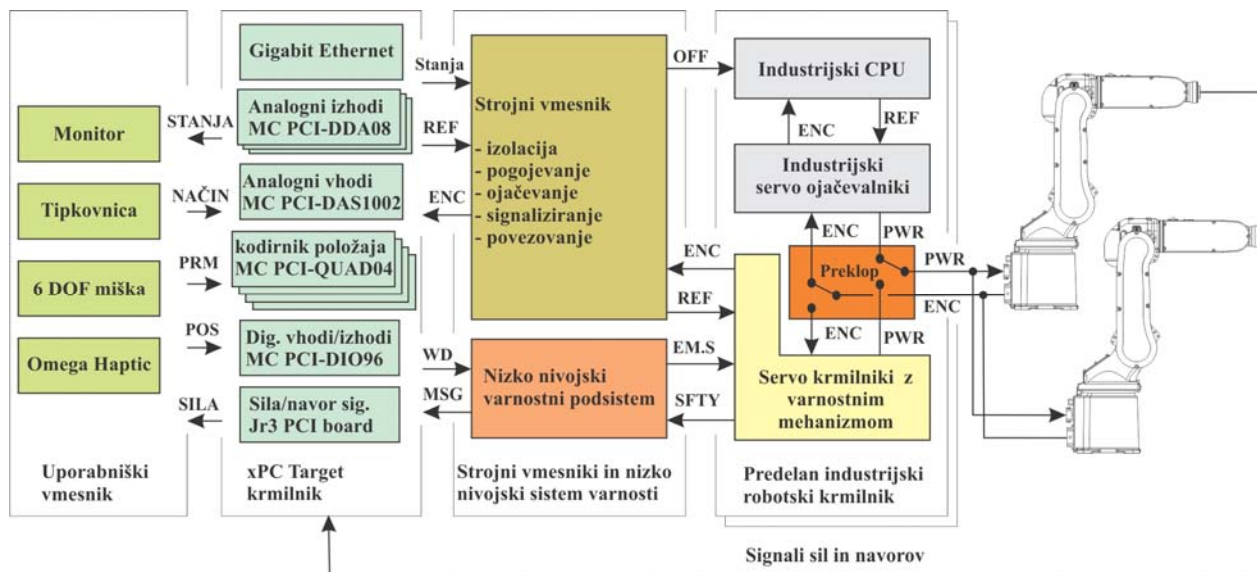
V primeru izgradnje xPC Target krmilnika za krmiljenje dveh Motoman-ovih manipulatorjev MH5 je bilo sodelovanje s proizvajalcem uspešno v tej smeri, da je bil pripravljen za nas predelati industrijski robotski krmilnik. Proizvajalec je za nas obstoječemu industrijskemu krmilniku, paralelno namestil 12 servo-krmilnikov s hitrimi odzivnimi časi in vgrajenim varnostnim mehanizmom (glej sliko 2). Izbiranje med industrijskim krmilnikom in dodanimi servo-krmilniki je izvedeno s preklapljanjem, s čimer je omogočeno le delovanje enega od njiju. Takšen namenski servo-krmilnik (glej sliko 2) omogoča nastavljanje parametrov, branje signalov, krmiljenje aktuatorjev posamezne osi manipulatorja in skrbi, da med njegovim delovanjem ne prihaja do napak, ki bi lahko poškodovale aktuator. Za nas je pomembno dejstvo, da ima servo-krmilnik veliko število vhodno/izhodno digitalnih in analognih linij preko katerih lahko nastavljamo in beremo različne parametre ter signale kodirnikov položaja. Poleg tega ima servo-krmilnik

možnost izbire krmiljenja navorov ali hitrosti z analognimi signali.



Slika 2: Servo krmilnik proizvajalca Yaskawa

V nadaljevanju članka je predstavljena generalna arhitektura strojne opreme (glej sliko 3), ki omogoča povezavo med xPC Target krmilnikom in servo-krmilniki obeh manipulatorjev. Ta je specifična in je razvita posebej za komuniciranje z Motoman-ovima robotoma MH5, zato bi morali pri izgradnji odprtega krmilnika kakega drugega robota te komponente odvzeti, dodati ali zamenjati. Generalna shema arhitekture je v celoti prikazana na sliki 3 in se sestoji iz petih sklopov komponent: dveh robotskih manipulatorjev MH5, predelanega industrijskega robotskega krmilnika (predelava je bila predhodno opisana), strojnih vmesnikov in nizko nivojskega sistema



Slika 3: Generalna arhitektura strojne opreme. Pomen okrajšav je: REF - krmilni signal za manipulator, PWR - napajalna linija za napajanje servo krmilnikov, ENC - signali kodirnikov položaja, EM.S - izklop v sili, MSG - sporočila varnostnega podsistema za xPC Target, WD - watchdog signal, STANJE - trenutno stanje krmilnika, NAČIN - uporabniška komanda, PRM - parametri definirani iz strani uporabnika.

varnosti (tu so bile vse elektronske komponente izdelane po meri in so namenjene preklapljanju med industrijskim krmilnikom in xPC Target krmilnikom), xPC Target krmilnika na katerem teče xPC Target operacijski sistem v realnem času ter glavnega računalnika, ki nam služi kot uporabniški vmesnik.

2.1 Izgradnja krmilnika z odprto arhitekturo

Na sliki 3 je prikazana celotna arhitektura strojne opreme narejene po meri, za predelan robotski krmilnik robotov, pri katerem so za krmiljenje manipulatorjev uporabljeni servo-krmilniki z analognimi vhodi. Pri razvoju in izbiri strojnih vmesnikov je potrebno paziti na to, da so signali industrijskega krmilnika in xPC Target krmilnika med seboj galvansko ločeni, da je omogočeno hkratno preklapljanje vseh signalov med industrijskim krmilnikom in xPC Target krmilnikom ter da je branje signalov kodirnikov položaja iz absolutnih senzorjev pravilno. Poleg omenjenih situacij je potrebno ves čas delovanja skrbeti tudi za varnost in ob najmanjši napaki prekiniti delovanje xPC Target krmilnika. Za vse omenjene težave smo razvili strojne vmesnike po meri in jih združili v sklop

strojnih vmesnikov in nizko nivojskega varnostnega sistema (glej sliko 3).

V primeru branja signalov kodirnikov položaja je potrebno paziti, da so signali kodirnikov položaja vseh dvanajstih servo-krmilnikov pravilno preusmerjeni k xPC Target krmilniku, v katerem so nameščene kartice za zajemanje signalov kodirnikov položaja MC PCI-QUAD04 (Measurement Computing Corporation, MC), ki omogočajo branje pozicij iz senzorjev. Za pravilno pošiljanje absolutnih vrednosti kotov v sklepih, je med oba krmilnika postavljena serijska komunikacija, ki med proceduro inicializacije skrbi za pridobivanje začetnih vrednosti kotov iz absolutnih kodirnikov položaja nameščenih v sklepih obeh manipulatorjev. Za pretvarjanje izračunanih krmilnih signalov hitrosti ali navora je uporabljenih več osem kanalnih 12 bitnih kartic MC PCI-DAS08 z analognimi vhodi. Pravilno krmiljenje in preklapljanje med kontrolnimi signali xPC Target krmilnika in signali industrijskega krmilnika je izvedeno s predhodno omenjenimi strojnimi vmesniki. Za vse digitalne linije namenjene branju in

nastavljanju operacijskih in varnostnih parametrov, so v xPC Target krmilniku uporabljene kartice MC PCI-DAS08 z 48 I/O digitalnimi linijami in MC PCI-DIO96H z 96 I/O digitalnimi linijami. Za zajemanje analognih signalov je uporabljena kartica MC PCI-DAS1002, ki ima 16 vhodnih kanalov. V xPC Target krmilniku se nahaja tudi JR 3 vmesniška kartica, ki jo uporabljamo za zajemanje podatkov sile/navora in pospeškov izmerjenih s senzorjem pritrjenim v prijemu robota.

Programiranje in razvoj krmilnika poteka v celoti na razvojnem računalniku z uporabniškim grafičnim vmesnikom. Ko je program na razvojnem računalniku razvit, ga prevedemo in preko mreže pošljemo xPC target krmilniku. Razvojni računalnik in xPC Target krmilnik ves čas delovanja komunicirata preko mrežne povezave, kar omogoča uporabniku nastavljanje vseh krmilnih signalov preko uporabniškega grafičnega vmesnika (glej sliko 3). Preko njega je mogoče tudi branje vseh sporočil, ki so povezana s kontrolo in varnostjo krmilnika. Na glavni računalnik je možno poleg običajne tipkovnice priklopiti še miško s šestimi stopnjami prostosti ali haptična robota Omega Haptic.

2.2 Varnostni mehanizmi

Zaradi sodelovanja manipulatorjev in človeka predstavlja varnostni mehanizem krmilnega sistema najpomembnejšo komponento pri izgradnji krmilnikov po meri. Še posebej pride varnostni mehanizem do izraza, ko sta človek in manipulator v fizičnem kontaktu in med seboj ves čas sodelujeta. V takšnih primerih mora biti zaradi zaščite človeka in opreme operacija varnosti zagotovljena v vsakem trenutku. V večini takšnih primerov je potrebno operacijo nizko nivojske varnosti zagotoviti mehansko in ne programsko.

Med razvijanjem krmilnikov po meri, moramo ves čas razvoja testirati in ovrednotiti strojno in programsko opremo, pri čemer smo običajno omejeni s sredstvi in časom. Zaradi tega si je potrebno pri izvedbi varnostnih

mehanizmov pomagati na najboljši možni način, glede na dana sredstva in čas. V našem primeru je varnost razdeljena na tri varnostne nivoje. Prvi varnostni nivo je implementiran na xPC Target krmilniku in bi v normalnih okoliščinah moral zaznati vse nepravilne situacije, ki se zgodijo med delovanjem xPC Target krmilnika. Vendar pa je resnici na ljubo ta nivo najbolj podvržen napakam in se nanj ne moremo vedno zanašati. Drugi varnostni nivo je implementiran v servo-krmilniku in skrbi, da med krmiljenjem aktuatorjev v sklepih manipulatorjev ne prihaja do napak. Servo-krmilnik ob napaki nemudoma ustavi krmiljenje aktuatorjev in pošlje sporočilo v obliki alarma. Tretji varnostni nivo je implementiran med xPC Target krmilnikom in servo-krmilniki ter je razvit po meri iz elektronskih komponent. Varnostni mehanizem ves čas delovanja primerja varnostne signale, ki signalizirajo pravilno delovanje servo-krmilnikov in xPC Target krmilnika. Ob napaki enega od teh signalov, mehanizem v hipu prekine z izvajanjem operacij in vklopi proceduro izklopa v sili.

Poleg vseh treh opisanih varnostnih mehanizmov je uporabniku na voljo tudi tipka za izklop v sili ali tipka za omogočanje delovanja. Pri prvi se ob pritisku nanjo delovanjem krmilnika v hipu preneha, medtem ko je potrebno imeti drugo za nemoteno delovanje krmilnika ves čas pritisnjeno.

3 Zaključek

Vsekakor je veliko vzrokov, ki govorijo v prid razvoja po meri narejenih krmilnikov, saj so standardni industrijski krmilniki običajno zaprti in uporabniku nedostopni. Ne glede na nekatere omejitve, ki jih kljub optimizaciji še imajo mehanske konstrukcije robotov, pa so prav krmilniki tisti, ki omejujejo razvoj in izgradnjo zahtevnejših robotskih aplikacij. Rešitev se pokaže v razvoju krmilnikov po meri, ki jih je možno optimizirati sebi v prid. S krmilniki po meri je možen hitrejši razvoj zahtevnejših nalog robota in razvoj aplikacij z integracijo senzorjev, ki delujejo v realnem

času. Z možnostjo razvijanja zahtevnejših naprednih aplikacij dobivajo roboti dodano vrednost, zaradi katere je možno takšne robote s pridom uporabiti tudi v izobraževalne namene. Po drugi strani pa se tudi v industrijskem okolju pojavlja vse več zahtevnih nalog, ki jih z obstoječimi industrijski krmilniki ni moč izvesti. Iz vsega tega lahko zaključimo, da je izgradnja in izbira prave strojne in programske opreme, ključna za hiter in uspešen razvoj kompleksnejših aplikacij.

4 Literatura

[1] T. Kröger, B. Finkemeyer, S. Winkelbach, S. Molkenstruck, L.-O. Eble, F. M. Wahl, "A Manipulator Plays Jenga", *IEEE Robotics and Automation Magazine*, pp. 79-84, 2008.

[2] F. Pertin, J. B. D. Tuves, "Real time robot controller abstraction layer", *Proceeding of the International Symposium on Robotics ISR 2004, Paris, France*, 2004.

[3] *High-speed Robot Synchronous Control Software Specifications*, Yaskawa Electric Corporation, Robotic Division, January 2009.

[4] J. Rejc, J. Cinkelj, M. Munih, "Dimensional measurements of a gray-iron object using a robot and a

laser displacement sensor", *Robotics and Computer-Integrated Manufacturing*, vol. 25, pp. 155–167, 2009.

[5] I. Verdenik, B. Šolar, Z. Balorda, B. Švab, "Robotski programski jezik LIRA", *Elektrotehniški vestnik*, 56, št. 1, str. 83-87, 1989.

[6] R. Kamnik, D. Matko, T. Bajd, "Application of model reference adaptive control to industrial robot impedance control", *J. intell. robot. syst.*, vol. 22, no. 2, str. 153-163., June 1998.

[7] A. Bardorfer, M. Munih, A. Zupan, A. Primožic, "Upper limb motion analysis using haptic interface", *IEEE/ASME Transactions On Mechatronics*, vol. 6, pp. 1–9, 2001.

[8] R. Riener, J. Hoogen, M. Ponikvar, R. Burgkart, M. Frey, G. Schmidt, "Orthopädischer trainingssimulator mit haptischem feedback (orthopaedic training simulator with haptic feedback)", *Automatisierungstechnik*, vol. 50, pp. 296–303, 2002.

[9] M. Ponikvar, J. Hoogen, M. Munih, G. Schmidt, R. Riener, "Haptic environment for analysis of smooth arm movements", *Proceedings of ICAR 2003, The 11th International Conference on Advanced Robotics*, Coimbra, Portugal, pp. 173–178, 2003.

[10] J. Podobnik, M. Munih, "Haptic interaction stability with respect to grasp force", *IEEE Transactions On Systems, Man, And Cybernetics-Part C: Applications And Reviews*, vol. 37, pp. 1214–1222, 2007.