

REKURZIVNI POSTOPEK TESTIRANJA VEČNIVOJSKEGA KOMUNIKACIJSKEGA SISTEMA

INFORMATICA 2/89

Keywords: multilevel communication system, testing, recursive procedure

Tone Vidmar, Jernej Virant
Fakulteta za elektrotehniko
in računalništvo, Ljubljana

P O V Z E T E K: Struktura, arhitektura in sintaksa testnega modela je povzeta po [1] in [13]. Za privzet univerzalni testni model je podan rekurzivni postopek testiranja večnivojskega komunikacijskega sistema (*npr. OSI referenčni model*) v njegovem implementacijskem okolju [13]. Postopek omogoča testiranje sistema od najnižjega komunikacijskega nivoja proti višjim. Predlagan rekurzivni algoritem zagotavlja uporabnost testne metode (*Perturbacija Globalnega Stanja Sistema PGSS*) [2], [3], [4], [5] pri testiranju poljubno kompleksnih komunikacijskih sistemov (*redukcija drevesa globalnih stanj ob večpasovnem testiranju*). Celoten postopek je formaliziran in programabilen kar omogoča, da se predlagana metoda razvije v razred t.i. implementacijskih generatorjev.

I. IZHODIŠČA

Glede na [1] je model komunikacijskega sistema opisan z mrežo končnih t.i. komunikacijskih avtomatov $E(i), E^*(i), m(i)$. Termin "komunikacijski avtomat" predstavlja običajen Mealyjev avtomat s specifično sintakso, ki bo opisana v nadaljevanju. $E(i)$ in $E^*(i)$ predstavljata model komunicirajočega para, medtem, ko $m(i)$ predstavlja model komunikacijskega medija v širšem smislu [6], [7]. Dogodki komunikacijskega avtomata so pošiljanje in sprejemanje komunikacijskih sporočil (*protokolarnih, servisnih, implementacijsko odvisnih in aplikacijskih [8]*). S stališča avtomatne teorije dogodki predstavljajo vhodno/izhodne abecede komunikacijskega avtomata. Dogodek v modelu se označuje z $\$ = \langle +, -, \# \rangle$. Komunikacijska sporočila, ki jih komunikacijski avtomat pošilja se označujejo z "-", tista pa, kijih sprejema se označujejo z "+". Lokalni dogodki, nič dogodki, se označujejo z "#". (*Predstavljajo interno funkcionalnost komunikacijskega avtomata.*). Sekvenco dveh dogodkov [6], [8]:

$$\begin{aligned} &(f_{ij}, +e_x(P_p), f_{ik}) \\ &(f_{ik}, -e_y(P_p), f_{il}) \end{aligned}$$

se v klasičnem Mealyjevem avtomatu interpretira kot, da je vhodna črka e_x/f_{ij} generirala izhodno črko e_y in povzročila prehod v stanje f_{il} . Sprejemno oddajna sekvenca bi Mealyevi notaciji ima naslednjo obliko $f_{ij}, e_x/e_y, f_{il}$. Podobna interpretacija je sprejemljiva za poljubne kombinacije dogodkov v komunikacijskem avtomatu. Tipe dogodkov [8] se lahko v smislu klasičnega Mealyjevega avtomata razdeli na množico vhodnih črk R ..receive- in množico izhodnih črk T ..transmit. Lokalni dogodek ustreza t.i. "c" prehodu [7], [8].

Vsak komunikacijski avtomat je torej mogoče interpretirati kot klasičen Mealyjev avtomat. Ta ugotovitev olajšuje nadaljnjo formalizacijo rekurzivnega postopka, ker se lahko privzame klasična avtomatna teorija [9], [10].

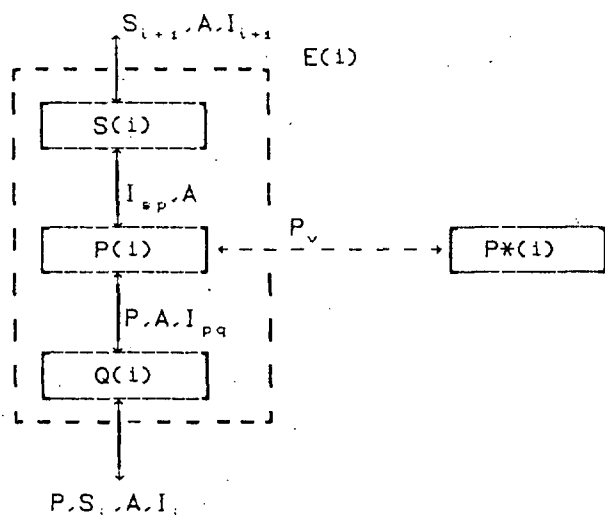
V splošnem se množici vhodno/izhodnih črk (* R ..vhodna abeceda T ..izhodna abeceda *) posameznih komunikacijskih avtomatov delita v podmnožico protokolarnih sporočil P , podmnožico servisnih sporočil S , podmnožico implementacijskih pogojev I in podmnožico aplikacijskih sporočil A . Po definiciji ISO - modela opravlja nižji nivo storitev za višji nivo, ki jo ta zahteva z pošiljanjem servisnih sporočil. Protokolarna sporočila predstavljajo tok virtualnega komunikacijskega kanala za protokolarni par $P(i)/P^*(i)$ na istem nivoju. I in A predstavljata realno okolje komunikacijskega nivoja. Za $E(i)$ in $E^*(i)$ velja, da morata vsebovati funkcionalnost protokola in obeh protokolarnih vmesnikov $S(i)$ in $Q(i)$ [6], [12]. Odnose omenjenih veličin vidimo na sliki 1.

Trojka $E(i), E^*(i), m(i)$ predstavlja mrežo med seboj povezanih komunikacijskih avtomatov:

$$\begin{aligned} E(i) &= \{ F_a, R_a, T_a, w_a, t_a \} = E_a \\ E^*(i) &= \{ F_b, R_b, T_b, w_b, t_b \} = E_b \\ m(i) &= \{ F_c, R_c, T_c, w_c, t_c \} = m_c \end{aligned}$$

ki jih nahajamo na sliki 2. Pri tem veljajo opredelitve:

- F - množica notranjih stanj komunikacijskega avtomata
 R - množica vhodnih črk komunikacijskega avtomata, vhodna abeceda
 T - množica izhodnih črk komunikacijskega avtomata, izhodna abeceda
 w - logična časovna funkcija prehajanja stanj
 t - logična funkcija izhodne črke



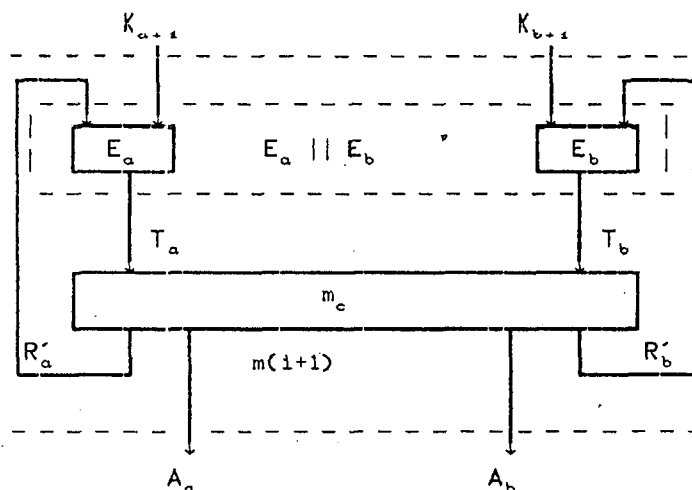
Slika 1. Rezina komunikacijskega nivoja ISO modela

II. REKURZIVNI POSTOPEK TESTIRANJA

V [2], [3], [4], [5] in [6] je opisana osnovna ideja testiranja protokolov z metodo PGSS. Opremljene so modifikacije metode, ki so pogojene z uvajanjem robnih pogojev v postopek testiranja [7] [8]. Vendar s tem in uvajanjem dodatnih sintaktičnih konstruktov v testni model problem ekspanzije števila globalnih stanj v drevesu globalnih stanj T še vedno ni zadovoljivo rešen. To je očitno na Sliki 3., podaja testni model sestavljen iz $2N+1$ komunikacijskih avtomatov. Testna metoda načelno sicer omogoča testiranje poljubnih mrež med seboj povezanih komunikacijskih avtomatov in tako tudi te, vendar pa sta vprašljiva izvedljivost in časovne performanse testiranja.

Gledano z določenega nivoja proti nižjim komunikacijskim nivojem, zajema prenosni medij v širšem smislu $m(i)$ fizični prenosni medij in vse nižje komunikacijske nivoje. To velja

za poljuben komunikacijski nivo, razen za prvi nivo. Za ta nivo obstaja kot model prenosnega medija samo $m(1)$, preko katerega komunicirata $E(1)$ in $E^*(1)$. Če se uspe trojko $E(1), E^*(1), m(1)$ nadomestiti z nekim novim avtomatom $m(2)$, je postopek ponovljiv na nivoju $E(2)/E^*(2)$ itd. Tako ponavljanje testiranja trojke je osnova rekurzivnega algoritma. Ob tem je topologija testnega modela vedno identična s topologijo, ki je podana na Sliki 2. Potrebno pa je poiskati način nadomeščanja trojke $E(i), E^*(i), m(i)$ z ekvivalentnim komunikacijskim avtomatom $m(i+1)$.



Slika 2. Avtomatna mreža modela

II.1. PARALELNO SERIJSKA KOMPOZICIJA MODELA

Na Sliki 2. vidimo, da se $m(i+1)$ lahko dobi s paralelno kompozicijo $E(i)$ in $E^*(i)$ v avtomat QP, ki ji sledi serijska kompozicija dobljenega komunikacijskega avtomata QP z $m(i)$. Ob ugotovitvi, da je komunikacijski avtomat izrazljiv z Mealyjevim avtomatom gre torej za problem paralelno serijske kompozicije takih avtomatov. Formalna osnova paralelno/serijske kompozicije je privzeta po [9] in [10]. Na Sliki 2 nahajamo naslednje relacije med abecedami posameznih komunikacijskih avtomatov:

$$\begin{aligned}
 T_a &= P_a \cup S_a \cup I_{ac} \cup A_b \\
 T_b &= P_b \cup S_b \cup I_{bc} \cup A_a \\
 T_c &= (P_a \cup I_{ca} \cup A_a) \times (P_b \cup I_{cb} \cup A_b) \\
 R'_a &= (P_a \cup I_{ca} \cup A_a) \\
 R'_b &= (P_b \cup I_{cb} \cup A_b) \\
 R_a &= R'_a \times K_{a+1} \\
 R_b &= R'_b \times K_{b+1} \\
 R_c &= T_a \times T_b \\
 K_{a+1} &= S_{a+1} \cup A_b \\
 K_{b+1} &= S_{b+1} \cup A_a
 \end{aligned}$$

A_a/A_a - množica aplikacijskih sporočil

P_a/P_b - množica protokolarnih sporočil, ki jih pošilja avtomat -a/b- in sprejme avtomat -b/a-.

S_a/S_b - množica servisnih sporočil avtomatov $E(i)/E*(i)$

S_{a+1}/S_{b+1} - množica servisnih sporočil višjih komunikacijskih nivojev

I_{ac}/I_{bc} - množica implementacijsko odvisnih sporočil avtomatov $E(i)/E*(i)$.

I_{ca}/I_{cb} - množica implementacijsko odvisnih sporočil avtomata $m(i)$

Struktura posameznih komunikacijskih avtomatov -a/b/c- je na osnovi zgornjih definicij sledeča:

$$E_a = (F_a, (P_a \cup I_{ca} \cup A_b) \times (S_{a+1} \cup A_b), (P_b \cup S_a \cup I_{ac} \cup A_b), w_a, t_a)$$

$$E_b = (F_b, (P_b \cup I_{cb} \cup A_a) \times (S_{b+1} \cup A_a), (P_a \cup S_b \cup I_{bc} \cup A_a), w_b, t_b)$$

$$m_i = (F_c, (T_a \times T_b), (P_a \cup I_{ca} \cup A_b) \times (P_b \cup I_{cb} \cup A_a), w_c, t_c)$$

Paralelna kompozicija komunikacijskih avtomatov -a/b- ima sledečo strukturo [9], [10]:

$$QP = ((F_a \times F_b), (R_a \times R_b), (T_a \times T_b), w_p, t_p)$$

$$w_p = (w_a(f_a, r_a), w_b(f_b, r_b))$$

$$t_p = (t_a(f_a, r_a), w_b(f_b, r_b))$$

$$f_a \in F_a$$

$$f_b \in F_b$$

$$r_a \in R_a$$

$$r_b \in R_b$$

Po serijski kompoziciji avtomata QP z komunikacijskim avtomatom -c- se dobi strukturo komunikacijskega medija v širšem smislu $m(i+1)$:

$$m(i+1) = QP \Leftrightarrow m(i)$$

$$m(i+1) = ((F_a \times F_b \times F_c), (R_a \times R_b), T_c, w_{i+1}, t_{i+1})$$

$$w_{i+1} = (w_a(f_a, r_a), w_b(f_b, r_b), w_c(f_c, (t_a, t_b)))$$

$$t_{i+1} = t_c(f_c, (t_a, t_b))$$

$$t_a \in T_a$$

$$t_b \in T_b$$

Dobljena algebrajska struktura je komunikacijski avtomat. Mreža treh komunikacijskih avtomatov $E(i), E*(i), m(i)$ je ekvivalentno nadomeščena avtomatom $m(i+1)$. Med posameznimi podmnožicami komunikacijskih sporočil tega komunikacijskega avtomata veljajo relacije in lastnosti, ki se morajo upoštevati v abecedah komunikacijskega avtomata $m(i+1)$:

L1: - Sporočila iz podmnožic $S_a, S_b, I_{ac}, I_{bc}, I_{ca}$ in I_{cb} se interno zaključijo v $m(i+1)$ in ne generirajo izhodne črke.

L2: - Sporočila iz podmnožic P_a in P_b se "skozi" proces -c- prenašajo transparentno od procesa -a- k procesu -b- in obratno.

L3: - Sporočila in podmnožic A_a in A_b se transparentno prenašajo skozi sistem.

Iz teh ugotovitev oziroma trditv sledi, da imajo za testiranje višjega komunikacijskega nivoja $E(i+1)/E*(i+1)$ relevantno vlogo le dogodki povezani z komunikacijskimi sporočili iz podmnožic S_{a+1}, S_{b+1}, A_a in A_b . Minimilirana logična struktura komunikacijskega avtomata $m(i+1)$, ki predstavlja za višji komunikacijski nivo komunikacijski medij v širšem smislu, se formalno dobi tako, da se vse črke iz abeced $P_a, P_b, S_a, S_b, I_{ac}, I_{bc}, I_{ca}$ in I_{cb} nadomesti z pogoji D.C. -Don't Care Condition- [10 str. 55, 119]. Dokaz o formalni upravičenosti takega postopka je privzet konceptualno, njegovo formalna osnova pa je podana v [9] na straneh 34, 88, 91 in 146. Postopek uvažanja pogojev D.C. ne zagotavlja optimalne kompresije notranjih stanj komunikacijskega avtomata $m(i+1)$ [10]. Na nekatere stranske efekte postopka redukcije notranjih stanj je opozoril J. Hartmanis v [14]. Za konkreten primer, ko gre za strukture elementa $m(i+1)$, se je izkazalo, da je pomembna zgolj korespondenca med stanji strukture $m(i+1)$ po postopku paralelno serijske kompozicije, z reduciranimi stanji, po uvedbi D.C. pogojev. $m(i+1)$ predstavlja servisni vmesnik komunikacijskega nivoja napram višjemu nivoju in model prenosnega medija, ki zagotovi transparentni prenos komunikacijskih sporočil višjega komunikacijskega nivoja. Ob upoštevanju predstavljenega koncepta je struktura komunikacijskega avtomata $m(i+1)$ sledeča:

$$m(i+1) = ((F_a \times F_b \times F_c), (S_{a+1} \cup A_b) \times (S_{b+1} \cup A_a), (A_a \times A_b), w_{i+1}, t_{i+1})$$

$$w_{i+1} = (w_a(f_a, (s_{a+1}, a_b)), w_b(f_b, (s_{b+1}, a_a)))$$

$$t_{i+1} = (t_c(f_c, (a_a, a_b)))$$

II.2. GENERACIJA VMESNIKA $S(i)$

Iz strukture mreže avtomatov ki jih nadomešča avtomat $m(i+1)$ je razvidno, da med posameznimi podmnožicami vhodno/izhodnih črk veljajo sledeče relacije:

$$R_{i+1} = (S_{a+1} \cup A_b) \times (S_{b+1} \cup A_a) = (S_{a+1} \times S_{b+1}) \cup (A_a \times A_b)$$

$$T_{i+1} = (A_a \times A_b)$$

$$R_{i+1} \cap T_{i+1} = S_{a+1} \times S_{b+1}$$

$$R_{i+1} \cup T_{i+1} = R_{i+1}$$

Se pred izpeljavo gornjih formalnih izrazov, se je pokazalo, da so intuitivno postavljene trditve L1, L2 in L3 na mestu.

Iz gornje razprave izhaja, da je funkcionalno gledano komunikacijski avtomat $m(i+1)$ hibridni model servisnega vmesnika $S(i)$ - in prenosnega medija v ožjem smislu $m(i)$.

Formalen dokaz za zgornjo trditev je mogoče zasnovati na postopku serijske dekompozicije [9], [10] avtomata $m(i+1)$ na komunikacijska avtomata $S(i)$ in $m(i)$. Postopek serijske dekompozicije ni trivialen in možen samo pod posebnimi pogoji. Formalno je definiran splošen primer serijske dekompozicije [10 str. 97, 137]. Serijska dekompozicija avtomata $m(i+1)$ je možna samo če obstoja particija Π_s , nortanje abecede F_{i+1} tako, da ima Π_s substitucijsko lastnost (*S.P.*) za F_{i+1} . S.P. predstavlja izraz [9]:

$$w_{i+1}(f_{i+1}, r_{i+1}) = w_{i+1}(f_s, r_{i+1})$$

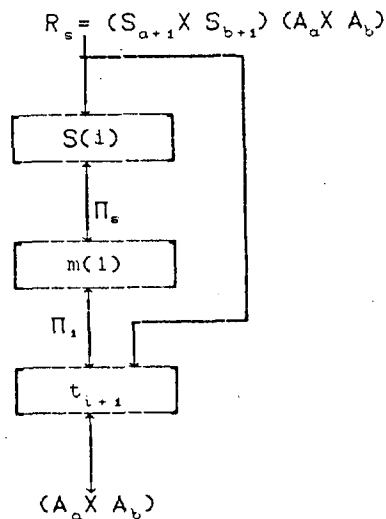
Za avtomata dobljena s tem postopkom velja:

$$S(i) = (\Pi_s, R_s, w_s)$$

$$m(i) = (\Pi_i, R_i, w_i)$$

$$t: \Pi_i \times R_s \rightarrow T = A_a \times A_b$$

Shematska prezentacija take dekompozicije je sledeča:



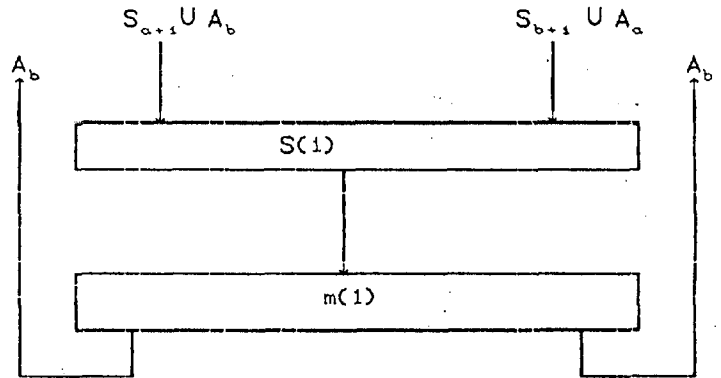
Slika 4.

Pogoj, ki zadošča za izpeljavo serijske dekompozicije je torej sledeč [9]:

$$\forall \exists ((\Pi_s \subset F_{i+1}, \Pi_i \subset F_{i+1}) \vee (\Pi_s \Pi_i = P_o) \vee (f_{i+1} = w_s(\Pi_s)) \vee (w_{i+1}(f_{i+1}, r_{i+1}) = w_{i+1}(f_s, r_{i+1})))$$

P_o = particija nič

Vprašanje pa je, ali ima $S(i)$ substitucijsko lastnost. Dokaz tega je lahko zasnovan na reverzibilnosti postopka kompozicije in dekompozicije [9]. Koncept dekompozicije kot formalen dokaz vsebine $m(i+1)$ potrjuje začetno trditev, da je rekurzivni postopek testiranja tudi aplikativni generator strukture servisnega vmesnika $S(i)$, katerega prikazuje slika 5.



Slika 5.

O opisanem postopku lahko govorimo kot o aplikacijskem generatorju, ker le ta na osnovi določenih vhodnih podatkov avtomatsko generira strukturo vmesnika, ki predstavlja implementacijsko odvisen del rezine komunikacijskega modela.

II.3. MINIMIZACIJA ABECED F , R , S AVTOMATA $m(i+1)$

Iz definicije globalnega stanja sistema, [1], [2], [3], [4] (matrika globalnega stanja) in drevesa globalnih stanj, kot rezultata perturbacijskega postopka, sledi, da je notranja abeceda avtomata $m(i+1)$ definirana z glavno diagonalo globalnih stanj drevesa globalnih stanj. Samo drevo globalnih stanj je reprezentacija avtomata, zato je na osnovi stanj čakalnih vrst, oziroma stranskih elementov matrike stanj, možno avtomatično generirati tudi abecedi R in T [13 str. 91]. Temu lahko sledi postopek minimizacije opisan v poglavju II.1. Tudi generacija abeced in posredno same strukture avtomata $m(i+1)$ je lastnost rekurzivne metode kot aplikativnega generatorja.

II.4. VPELJAVA VMESNIKA $Q(i+1)$

Vpeljava vmesnika $Q(i+1)$ je vezana na strukturo komunikacijskega nivoja prikazanega na sliki 1. Vpeljava je ekvivalentna problemu modeliranja protokola $P(i+1) \setminus P^*(i+1)$ na osnovi

neformalne specifikacije s tem, da je za vmesnik $Q(i+1)$ znan vmesnik $S(i)$. Isto velja tudi za uvažanje implementacijskih pogojev bodisi na nivoju avtomatov E_a in E_b bodisi na nivoju m_c . Opirati se treba predvsem na strukturo in sintakso testnega modela [8], vse ob upoštevanju robnih pogojev testiranja [8], [13].

III. ZAKLJUČEK

Članek podaja formalno osnovo rekurzivnemu algoritmu -RECALG- večnivojskih komunikacijskih sistemov. Za N - nivojski komunikacijski sistem je struktura rekurzivnega algoritma sledeča:

```

RECALG
  i=1
  MODELIRANJE  $m(1), S(1)$ 
  DOKLER  $i \leq N$ 
    MODELIRANJE  $E(i), E^*(i), Q(i)$ 
    DOKLER TEST NIVOJA NI POZITIVEN
      GENERACIJA DREVEŠA GLOBALNIH STANJ
    KONEC
    GENERACIJA ABECED  $F_{i+1}, R_{i+1}, S_{i+1}, m_{i+1}$ 
    MINIMIZACIJSKI POSTOPKI PROCESA  $m(i+1)$ 
    GENERACIJA  $S(i)$ 
     $i = i + 1$ 
  KONEC
RECALGEND

```

Praktičen primer izvajanja algoritma je podan v [13]. Opis uporabljene programske opreme pa v [6]. Metoda omogoča "držati" dinamične strukture podatkov, ki se tvorijo v procesu perturbiranja, v implementacijsko obvladljivem obsegu, tudi pri testiranju velikih realnih sistemov. Metoda je bila preverjena in polavtomatsko simulirana na velikem realnem telekomunikacijskem sistemu. V nadaljevanju dela želimo uporabnost metode razširiti za poljubne topologije med seboj povezanih procesov, obstoječo pa avtomatizirati.

LITERATURA:

- [1] Pitro Zafiropulo, "Protocol Validation by Duologue-Matrix Analysis", IEEE Transactions on Communications, Vol. Com-26, december, 1980.
- [2] Pitro Zafiropulo, Colin H. West, Harry Rudin & Daniel Brand, "Towards Analysing and Synthesizing Protocols", Transactions on Communications, Vol. Com-28, April 1980.
- [3] Harry Rudin, Colin H. West, "A Validation Technique for Tightly Coupled Protocols", IEEE Transactions on Computers, Vol. C-31, July 1982.
- [4] Harry Rudin, "Automated Protocol Validation: Some Practical Examples", Proceedings of The Sixth International Conference on Computer Communications Protocol Validation, IBM J. RES. Vol. 22, July 1978.
- [5] Colin H. West, "General Technique for Communication Protocol Validation", IBM J. RES. Vol. 22, July 1978.
- [6] Tone Vidmar, "Modeliranje komunikacijskih protokolov s končnimi avtomati", Mipro 88, maj 1988, Opatija.
- [7] Tone Vidmar, "Logični protokolarni analizator", Mipro 88, maj 1988, Opatija.
- [8] Tone Vidmar, "Sintaksa medprocesnega komuniciranja", ETAN 88, junij 1988, Sarajevo.
- [9] J. Hartmanis, R. E. Stearns, Algebraic Structure Theory of Sequential Machines, Prentice - Hall 1966.
- [10] Jernej Virant, Preklopne funkcije, strukture in sistemi, Fakulteta za elektrotehniko 1985.
- [12] Tomaž Kalin, Tone Vidmar, "Logično testiranje protokolov", Informatica Vol. 1, Ljubljana januar 1988.
- [13] Tone Vidmar, Doktorska disertacija, Fakulteta za elektrotehniko, Ljubljana maj 1987.
- [14] J. Hartmanis "Further Results on the Structure of Sequential Machines", Journal of the Association for Computing Machinery, Vol 10, No. 1, January 1963, 78-88.