SPOLEHLIVOST PROGRAMOVÉHO VYBAVENÍ PRO OBVODY VYSOKÉ INTEGRACE A OBVODY VELMI VYSOKÉ INTEGRACE

Artem GANIYEV[1], *Jan VITÁSEK*[1]

[1]Katedra telekomunikační techniky, Fakulta elektrotechniky a informatiky, VŠB-TU Ostrava, 17. listopadu 15, 708 33 Ostrava-Poruba, Česká republika

artem.ganiyev.st@vsb.cz, jan.vitasek@vsb.cz

Abstrakt. *Článek se zabývá zpracováním metodiky hodnocení bezporuchovosti obvodů vysoké integrace (LSI) a velmi vysoké integrace (VLSI). V článku je provedena srovnávací analýza faktorů určujících bezporuchovost integrovaných obvodů, analýza existujících metodik a modelu hodnocení bezporuchové činnosti obvodů vysoké integrace a velmi vysoké integrace. Stěžejní částí článku je popis navrženého algoritmu a programu pro analýzu poruchovosti obvodů vysoké integrace a velmi vysoké integrace.*

Klíčová slova

Obvod vysoké integrace, obvod velmi vysoké integrace, paměťový prvek, paměťové zařízení, programové vybavení, telekomunikační systém.

1. Úvod

V tomto článku je zpracována metodika hodnocení bezporuchovosti paměťových obvodů vysoké integrace a velmi vysoké integrace radiotechnických systémů (RTS).

Zdokonalování současných RTS včetně telekomunikačních systémů je spojeno s vývojem současných přístrojových a informačních technologií. Hlavními směry jejich rozvoje jsou: využití číslicových technologií, zmenšení rozměrů, zvýšení stupně integrace.

Vývoj současných systémů radiových spojů (mobilní komunikační zařízení, digitální televize, digitální rozhlas a radionavigační systémy atd.) byl umožněn zvýšením množství funkcí, stupně integrace, složitosti.

LSI a VLSI obvody jsou rovněž základem všech výpočetních systémů, které se používají ke kontrole a řízení technologických procesů, kontrole a řízení RTS, kde bezporuchovost těchto systémů závisí na bezporuchové funkci LSI a VLSI obvodů.

Existují různé metody zvyšování bezporuchovosti LSI a VLSI obvodů, avšak otázkám přístrojové a programově-přístrojové redundance pro zvýšení bezporuchovosti LSI a VLSI jako základní metodě zlepšování jejich provozuschopnosti se nevěnuje dostatečná pozornost. V některých případech se rovněž nehodnotí jejich bezporuchovost při přístrojové redundanci. Obvykle se hodnotí bezporuchovost LSI a VLSI se zřetelem k nenadálým poruchám a v některých případech se zohledňuje počátek poruchy, ale zpravidla se nevěnuje pozornost spolehlivosti programového vybavení (PV), které je v současných výpočetních systémech RTS jejich neoddělitelnou součástí. Úkol zpracování komplexní metodiky hodnocení bezporuchovosti obvodů LSI a VLSI je proto aktuální.

Cílem práce je zpracování metodiky hodnocení bezporuchovosti obvodů LSI a VLSI se zřetelem k technologickým poruchám a spolehlivosti PV a návrh programu pro jejich výpočet.

K dosažení stanoveného cíle bylo nutné řešit následující úkoly:

- provést srovnávací analýzu faktorů určujících bezporuchovost integrovaných obvodů,
- provést analýzu existujících metodik hodnocení bezporuchovosti funkčnosti LSI a VLSI,
- provést analýzu metod zvyšování bezporuchovosti LSI a VLSI a zpracovat návrhy na zvýšení jejich spolehlivosti,
- prozkoumat varianty z hlediska bezporuchovosti integrovaných obvodů různého stupně integrace vzhledem k přístrojové a programově-přístrojové redundanci,
- provést analýzu modelů hodnocení spolehlivosti PV LSI a VLSI z hlediska technologických faktorů a navrhnout odpovídající metodiku hodnocení spolehlivosti programového vybavení,
- zhodnotit efektivitu zpracovaných metodik.

Při řešení stanovených úkolů byly uplatněny analytické a programové metody výzkumu na základě zpracovaných metodik s následným zpracováním a analýzou.

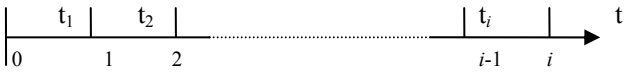
2. Teorie

2.1 Volba modelu pro hodnocení spolehlivosti programového vybavení

Jak již bylo poznamenáno, hodnocení a prognostika spolehlivosti programového vybavení (PV) se provádí na základě matematických modelů spolehlivosti programů.

Na časové přímce Obr. 1 je dole uvedeno číslo poruchy programu a nahoře jsou uvedeny časové intervaly mezi jednotlivými poruchami programu [1].

Představme si, že ve výchozím čase program pracuje a zachová svou funkčnost do konce časového intervalu t_1, kdy se objeví první porucha programu.



Obr. 1: Momenty poruch programu v čase.

Programátor pak „opraví‘‘ program, který poté pracuje správně v čase t_2 atd. Ať je náhodný čas mezi poruchou i a (i+1) funkcí hustoty rozložení f (ti), kde parametr λ_i – míra intenzity poruch. Čím je λ_i menší, tím kvalitnější je program. Je nutné zmenšit λ_i, tj. vztah $\lambda_i < \lambda_{i-1}$ pro všechna i. Rozebereme některé matematické modely spolehlivosti PV, proanalyzujeme je a zvolíme odstranit první makromodel.

1) Model podle Jelinského – Morandy

Nejprostší model klasického typu. Jeho základem jsou následující předpoklady [8]:

1. intenzita zjištění chyb R *(t) proporcionálně* průběžnému počtu chyb v programu, tj. počtu zbývajících (prvotních) chyb po odpočtu chyb odhalených,
2. všechny chyby jsou stejně důvěryhodné a jejich odhalení vzájemně nesouvisí,
3. čas další poruchy je rozdělen exponenciálně,
4. chyby se stále korigují, aniž by byly vneseny nové,
5. intenzita chyb nebo pravděpodobnosti rizika R *(t)=const* v intervalu mezi dvěma sousedními momenty výskytu chyb.

Rovnice rizikovosti je následující:

$$R(t) = K(B - (i-1)), \tag{1}$$

kde

- t – libovolný bod času mezi odhalením chyb (i-1) a i,
- K – neznámý koeficient proporcionality,
- B – výchozí prvotní (neznámý) celkový počet chyb, které zůstávají v PV.

V souladu s těmito předpoklady se pravděpodobnost bezporuchového chodu programů vyjádří jako funkce času t_i.

$$\begin{aligned} P(t_i) &= e^{-\lambda_i t_i} \\ \lambda_i &= R(t_i) \end{aligned}. \tag{2}$$

2) Šumanův model

Tento model se liší od modelu Jelinského-Morandy tím, že jsou časové etapy odladění a provozování hodnoceny zvlášť [2].

Model spočívá na následujících předpokladech:

1. celkový počet příkazů v programu je ve strojovém jazyce stálý,
2. na počátku testování se počet chyb rovná určité stálé veličině a úměrně opravě chyb jejich počet klesá, v průběhu testování programu nové chyby nevznikají,
3. chyby jsou již na počátku zjistitelné a podle celkového počtu opravených chyb lze hodnotit zbývající,
4. intenzita poruch programu je proporcionální počtu zbývajících chyb.

Při využití těchto předpokladů získáváme intenzitu odhalení poruch na časovém intervalu t (frekvence objevení chyb):

$$\lambda_s(t) = K_s e_r(\Delta\tau), \tag{3}$$

kde

- t – doba provozování systému,
- K_s – koeficient proporcionality-úměrnosti,
- $e_r(\Delta\tau)$ – počet zbývajících chyb v čase τ, ve vztahu k celkovému počtu příkazů I,
- $\Delta\tau$ – doba ladění programu odpočítávaná od momentu počátku sestavení systému programového zajištění.

Pravděpodobnost bezchybného provozování:

$$p(t) = \exp\left[-\int_0^t \lambda_s(\Delta\tau)d\tau\right] = \exp\left[-\int_0^t K_s e_r(\Delta\tau)d\tau\right]. \tag{4}$$

Pokud se v daném modelu frekvence výskytu chyb pokládá za nezávislou na čase t, je pokládána za stálou a následně je střední doba bezporuchové funkce programu rovna:

$$T_0 = \frac{1}{\lambda_s(t)} = \frac{1}{K_s e_r(\Delta\tau)}. \tag{5}$$

Takto lze využít makromodely pro zpracování PV ke zhodnocení spolehlivosti PV model Šumanův, vyžadující znalost zbývajících chyb a tyto lze odhalit Holstedovou metrikou. Popis a podstatu realizace této metriky – viz dále.

2.2 Modely zpracování programového vybavení

Jak bylo uvedeno v kapitole 2.1, ke zhodnocení spolehlivosti PV lze použít Holstedovu metriku. Podle této hypotézy se celkový počet určovaných chyb B (předávaných uživateli) v programu určuje podle složitosti jejího sestavení a tento program lze charakterizovat jako programovací činnost E nebo podle velikosti programu V [4], [6]:

$$B \approx CE ,$$

kde C je koeficient proporcionality.

Potencionální rozsah programu (v bitech) [7]:

$$V^* = \left(n_2^* + 2\right)\log_2\left(n_2^* + 2\right), \qquad (6)$$

kde n_2* – minimální počet různých operandů, které přestavuje obyčejně počet volných vstupních a výstupních parametrů. Hodnota n_2 může být určena ještě ve stádiu technického zadání zpracování PV.

V takovém případě je velikost programu V rovna [7]:

$$V = \left(V^*\right)^2 / \gamma , \qquad (7)$$

kde γ – úroveň jazyka programování (metrika).

Pro jazyk na úrovni strojového kódu $\gamma = 0.88$, pro jazyk programování PL1 $\gamma = 1.53$, pro anglický jazyk (jazyk vyšší úrovně) $\gamma = 2.16$.

Programovací činnost lze vyjádřit následovně [7]:

$$E = V^2 / V^* = \left(V^*\right)^3 / \gamma^2 = V / L , \qquad (8)$$

kde $L = V^* / V$ – úroveň programu. Pro jazyk velmi vysoké úrovně je $E = 3000$ [7].

Počet chyb v programu lze určit rovnicí [7]:

$$B = \frac{\left(V^*\right)^2}{3000\gamma} = \frac{V}{3000} . \qquad (9)$$

Při použití Šumanova modelu ke zhodnocení PV a výrazu (9) lze určit základní kritéria spolehlivosti PV.

Intenzita poruch PV:

$$\lambda_{PV}(t) = K_s e_r(\Delta\tau) = K_s B_r , \qquad (10)$$

kde

- t – doba činnosti programu,
- K_s – koeficient proporcionality,
- $B_r = e_r(\Delta\tau)$ – počet zbývajících chyb B v časovém okamžiku τ vztahující se k celkovému počtu příkazů I, tj. $B_r = B/I$,
- $\Delta\tau$ – doba odladění programu odpočítávaná od momentu sestavování systému PV.

Pravděpodobnost bezporuchové činnosti PV při použití exponenciálního zákona rozdělení poruch času výskytu chyb:

$$P(t) = \exp(-\lambda_{PV} t) = \exp(-K_s B_r t) . \qquad (11)$$

3. Zpracování algoritmů a programů pro hodnocení spolehlivosti PV paměťových zařízení

3.1 Statistický model hodnocení spolehlivosti programového vybavení

V případě, že je známa intenzita poruch LSI (VLSI), lze navrhnout a hodnotit spolehlivost PV podle statistických údajů, tj. technických (katalogových) údajů konkrétních LIS (PZ). Na základě statistických údajů [3] uvedených v Tab. 1 – 3 a Obr. 3 – 5 (k.2.4.) lze navrhnout následující matematický model intenzity poruch PV λ_{PV}:

$$\lambda_{PV} = K_1 \lambda_{PZ} - \lambda_{kor} , \qquad (12)$$

kde

- λ_{PZ} – intenzita poruch LIS PZ,
- λ_{kor} – intenzita poruch LIS PZ s korigováním kódových slov; K_1 – koeficient proporcionality.

Uvedeme konkrétní algoritmy a programy komplexního zhodnocení spolehlivosti LIS (VLIS) a PV pro daný model se zřetelem k podmínkám provozování.

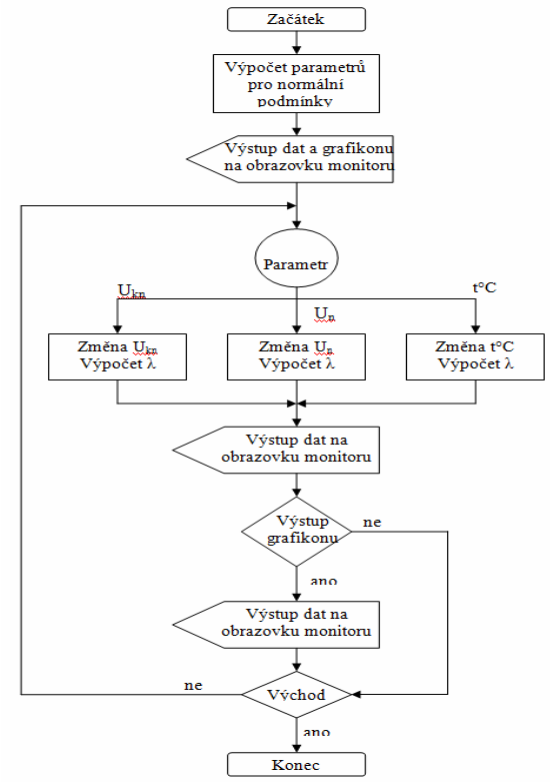
3.2 Zpracování programu zhodnocení spolehlivosti LIS (VLIS) paměťové zařízení a programového vybavení z hlediska podmínek provozování

Na Obr. 2 je uveden algoritmus programu hodnocení spolehlivosti LIS (VLIS), paměťového zařízení (PZ) a programového vybavení (PV) z hlediska podmínek k provozování.

Program pracuje následovně:

1. výpočet hodnot parametrů provozování pro normální režim funkčnosti VLIS paměťových zařízení,

2. na monitoru vystoupí výsledky výpočtů a grafikonu závislosti P (t),
3. probíhá volba parametru provozování a definice jeho významu,
4. proběhne změna zvoleného parametru,
5. vypočítávají se hodnoty intenzity poruchy LIS (VLIS) PZ pro zvolenou hodnotu parametru,
6. výsledky výpočtů vystupují na monitoru,
7. v případě nutnosti probíhá výstup grafikonu závislosti P (t) na monitoru,
8. jestli je experiment ukončen, vystoupí z daného bloku programu.



Obr. 2: Algoritmus programu hodnocení spolehlivosti provozování LIS (VLIS) PZ a PV z hlediska podmínek provozování.

V souladu s uvedeným algoritmem byl vypracován program v programovacím jazyce Turbo Pascal (verze 7.0) umožňující sledovat vliv provozních faktorů na spolehlivost LIS (VLIS) PZ a PV. Program umožňuje změnit ty parametry, jimiž jsou např. napětí kritického náboje U_{kn}, napětí napájení U_n a teplotu okolního prostředí T. Program umožňuje sledovat vliv současně jen jednoho faktoru.

Při výpočtech se používá vztah:

$$\frac{T_2}{T_1} = \sqrt{\frac{\pi}{2}N}\,\frac{n_i}{n_0} = 1.25\,\frac{n_i}{n_0}\sqrt{N}\,, \qquad (13)$$

kde:

- $T_2 = \frac{1}{\lambda_0 n_0}\sqrt{\frac{\pi}{2N}}$ – výraz zhodnocení pro průměrnou dobu vyhledávání matice LIS PZ a korigování ve stavu provozování,
- λ_0 – intenzita poruch paměťového prvku mikroschémat (10-9…10-11 1/hod),
- n_0 – délka slova při použití korigovaní,
- N – počet slov v paměťovém zařízení,
- n_i – délka kódového slova bez korigovaní,
- $T_1 = 1/(\lambda_0 N n_i)$ – průměrná doba vyhledaní matice LIS PZ bez korigovaní ve stavu provozovaní.

Při zpracování programu byly požity následující výchozí údaje (data):

1. použije se LIS PZ,
2. použije se Hemmingův kód korigování.

Pro LIS PZ disponuje následujícími technickými daty [5]:

- λ_0 - intenzita poruch paměťových prvku (PP) mikroschémat, $\lambda_0 = 1{,}556 \cdot 10^{-10}$ 1/hod,
- n_0 - délka slova při použití korigovaní, $n_0 = 15$,
- N - počet slov v PZ, $N = 262144$,
- n_i - délka kódového slova bez korigovaní, $n_i = 8$,
- $T_1 = 1/(\lambda_0 N n_i)$ – průměrná doba vyhledaní matice LIS PZ bez korigování ve stavu provozovaní.

Normální podmínky pro provozování daného LIS PZ jsou U_{kn} = 2,5 V, U_n = 5 V ± 5 %, T = 22 °C [5]. Program umožňuje provádět výzkum vlivu změny těchto parametrů na spolehlivost LIS PZ a PV. U_{kn} se mění od 1,2 V do 2,4 V; U_n – od 4,5 V do 5,5 V; T – od 18,5 °C do 44,5 °C.

Pro výše uvedená data získáváme následující hodnoty průměrného času matice LIS PZ bez korigování v provozuschopném stavu:

- T_1 = 3065 hodin,
- T_2 = 1048791 hodin,
- T_2/T_1 = 342,2.

Získané údaje umožňují vyvodit, jak použití Hemmingova kódu korigování zvyšuje průměrnou dobu (čas) vyhledávání matice LIS PZ v provozuschopném stavu 342,2 krát. Určíme základní charakteristiky velikosti spolehlivosti LIS PZ.

Intenzita poruch matice PZ: $\lambda_m = 1/T$. Pro PZ bez korigování chyb:

$$\lambda_{m1} = 1/T_1 = 1/3065 = 3.263 \cdot 10^{-4}\ (1/\text{hod}),$$

a s korigováním chyb:

$$\lambda_{m2} = 1/T_2 = 1/1430170 = 9.535 \cdot 10^{-7}\ (1/\text{hod}).$$

Poté představuje pravděpodobnost provozování $P_1(t) = e^{-\lambda t}$ bez korigování chyb za 24, 1000 a 10000 hodin:

$$P_1(24) = e^{-3{,}263 \cdot 10(-4) \cdot 24} = 0{,}9922,$$
$$P_1(1000) = e^{-3{,}263 \cdot 10(-4) \cdot 1000} = 0{,}7216,$$
$$P_1(10000) = e^{-3{,}263 \cdot 10(-4) \cdot 10000} = 0{,}0383.$$

Pro PZ a korigování chyb:

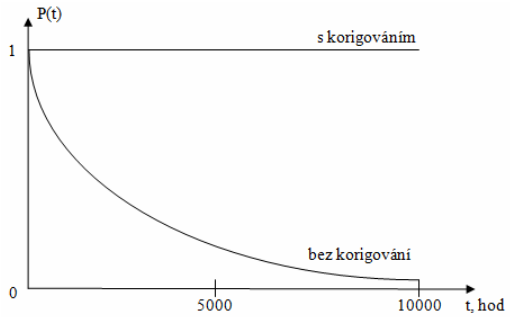
$$P_2(24) = e^{-6.99 \cdot 10(-7) \cdot 24} = 0{,}9999,$$
$$P_2(1000) = e^{-6.99 \cdot 10(-7) \cdot 1000} = 0{,}9991,$$
$$P_2(10000) = e^{-6.9 \cdot 10(-7) \cdot 10000} = 0{,}9905.$$

Získané výsledky ukazují, že použití Hemmingova kódu korigování umožňuje značně zvýšit spolehlivost uchovaných informací, v daném případě programového vybavení. Výsledky výpočtů intenzity poruch PZ jsou uvedeny v Tab. 1 a na Obr. 3 – grafikon závislosti P (t) pro normální podmínky (U_{kn} = 2,5 V, U_n = 5 V, T = 22 °C). V Tab. 2 je uveden výpočet intenzity poruch PZ a na Obr. 4 – grafikon závislosti $P(t)$ při U_{kn} = 2,5 V, U_n = 5 V, T = 30,5 °C. V Tab. 3 je uveden výpočet intenzity poruch PZ a na Obr. 5 – grafikon závislosti $P(t)$ při U_{kn} = 2,5 V, U_n = 5 V, T = 44,5 °C.

Tab.1: Výpočet intenzity poruch PZ, U_{kn} = 2,5 V, U_n = 5 V, T = 22 °C.

Ukazatelé spolehlivosti PZ	Bez korigování	S korigováním
Průměrná doba vyhledávání v provozuschopném stavu, hod	3065	1048791
Intenzita poruch RAM, 1/hod	$3{,}262642 \cdot 10^{-4}$	$9{,}534788 \cdot 10^{-7}$
Pravděpodobnost bezporuchového provozování za 24 hod	0,99220	0,99998
Pravděpodobnost bezporuchového provozování za 1000 hodin	0,72161	0,99905
Pravděpodobnost bezporuchového Provozování za 10000 hod	0,03829	0,99051
Intenzita poruch PV, 1/hod	3,2531E-04	
Vztah časů, s korigováním/bez korigování	342,2	

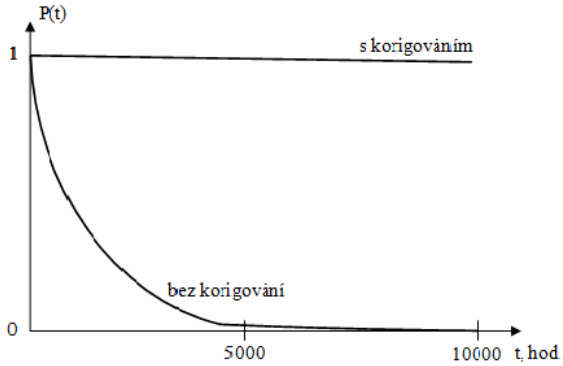
Pro snadnou srovnatelnost pravděpodobností bezporuchového provozování v různých podmínkách při použití kódu korigování i bez něho utváří program grafikon závislosti pravděpodobnosti bezporuchového provozování na čase $P(t)$ (Obr. 3).



Obr. 3: Grafikon závislosti $P(t)$ pro normální podmínky (U_{kn} = 2,5 V, U_n = 5 V, T = 22 °C).

Tab.2: Výpočet intenzity poruch PZ, U_{kn} = 2,5 V, U_n = 5 V, T = 22 °C.

Ukazatelé spolehlivosti PZ	Bez korigování	S korigováním
Průměrný čas výskytu v provozuschopném stavu v hod	1 024	350498
Intenzita poruch RAM 1/hod	$9{,}76563 \cdot 10^{-4}$	$2{,}85308 \cdot 10^{-6}$
Pravděpodobnost bezporuchového provozování za 24 hod	0,97684	0,99993
Pravděpodobnost bezporuchového Provozování za 1 000 hodin	0,3766	0,99715
Pravděpodobnost bezporuchového Provozování za 10 000 hodin	0	0,97187
Intenzita poruch PV, 1/hod	$9{,}7371 \cdot 10^{-4}$	
Vztah časů, s korigováním/bez korigování	342,3	

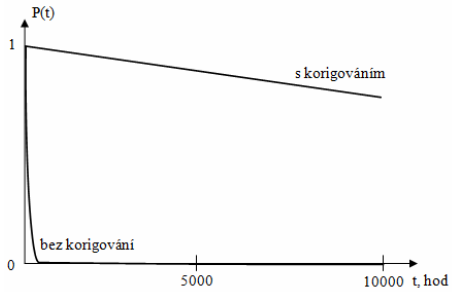


Obr. 4: Grafikon závislosti $P(t)$ pro normální podmínky (U_{kn} = 2,5 V, U_n = 5 V, T = 30,5 °C).

Tab.3: Výpočet intenzity poruch PZ, U_{kn} = 2,5 V, U_n = 5 V, T = 22 °C.

Ukazatelé spolehlivosti PZ	Bez korigování	S korigováním
Průměrný čas výskytu v provozuschopném stavu v hod	127	43453
Intenzita poruch RAM 1/hod	$7{,}8740 \cdot 10^{-3}$	$2{,}30134 \cdot 10^{-5}$

Pravděpodobnost bezporuchového provozování za 24 hod	0,82781	0,99945
Pravděpodobnost bezporuchového Provozování za 1 000 hodin	0,00038	0,97725
Pravděpodobnost bezporuchového Provozování za 10 000 hodin	0	0,79443
Intenzita poruch PV, 1/hod	$7{,}85110^{-3}$	
Vztah časů, s korigováním/bez korigování	342,1	



Obr. 5: Grafikon závislosti $P(t)$ pro normální podmínky (U_{kn} = 2,5 *V*, U_n = 5 *V*, T = 44,5 °*C*).

4. Závěr

Na základě provedených výzkumů lze stanovit:

1. Analýza existujících metod kontroly LIS a VLIS ukázala, že nejefektivnější je funkčně-parametrická metoda kontroly kombinující schémata s atestací a takto lze nejúčinněji dosáhnout zvýšení bezporuchovosti a spolehlivosti získávaných informací.
2. Byly prověřeny a navrženy modely umožňující hodnotit celkovou bezporuchovost LIS a VLIS při různých kombinacích zákonů klasifikace poruch vznikajících v LIS a VLIS.
3. Byl navržen model umožňující určit pravděpodobnost současného vzniku poruch ve výpočetním systému s vestavěným kontrolním systémem.
4. Byla navržena metodika hodnocení ekonomického přínosu zpracované metodiky diagnostikování LIS podle testovacího programu.
5. Bylo uvedeno, že pro technologii makromodelového zpracování programového vybavení je Šumanův model hodnocení spolehlivosti programového vybavení nejefektivnější.
6. Bylo uvedeno hodnocení spolehlivosti programového vybavení pro LIS (VLIS) pro případy s korigováním i bez něj a bylo zjištěno, že: použití kódu korigování zvyšuje bezporuchovost komplexu LIS (VLIS) – programového vybavení; použití Hammingova kódu v režimu korigování umožňuje zvýšit čas (průměrnou dobu) provozuschopného stavu komplexu LIS (VLIS) – PV až 340 krát.

Poděkování

Děkuji panu prof. Ing. Divišovi, CSc. a Ing. Tesařovi za účinnou a odbornou pomoc a další cenné rady. Příspěvek vznikl za aktivní podpory SGS SP/2010185 a GA 102/09/0550.

Použitá literatura

[1] ZEBEME, M. A. *Efektivnost razrabotki instrumentalnych sredstv dla telekommunikacionnogo programmnogo obespečenia*. Elektrosvaz, 2000. No. 7. – s. 33-35.

[2] KALABEKOV, B. A. *Cifrovie ustrojstva a mikroprocessornie systemy:* - *M*. Goračaja linia – Telekom, 2000. 336 s.: il.

[3] DERUGIN, A. A.; CIRKIN, V. V.; KRASOVSKIJ, V. E.; Pod red. GORDONOVA, A. U.; DERUGINA, A. A. – M. *Primenenie integrálních mikroschémat paměti: Spravočnik*: Radio a svaz, 1994. 232 s.: il.

[4] ABDULAEV, D. A. *Potoki izmeritelnoj informaci v systemach s postojannoj častotoj oprosa*. Izvestia Akademii nauk USSR. Seria techničeskych nauk. No. 1., 1970. s. 3-7.

[5] LEBEDEV, O. N. *Mikroschemy pameti a ich primenenie. – M.* Radio a svaz, 1990. 160 s. (Massovaja radiobiblioteka; Vyp. 1152).

[6] KEJDŽAN, G. A. *Prognozirovanie nadežnosti mikroelektronnoj apparatury na baze LIS. – M.* Radio a svaz, 1987.

[7] IYUDU, K. A. *Nadežnost, kontrol a diagnostika vyčíslitelných mašin a systém*. 1989. s. 45-51s.

[8] HLAVIČKA, J.: *Diagnostika a spolehlivost*, skripta ČVUT Praha, 1998. ISBN 80-01-01846-6.

O autorech

Artem GANIYEV se narodil v Taškentu. Titul Ing. získal na fakultě Elektrotechniky a komunikačních technologií v Ostravě studiem oboru Elektronika a sdělovací technika v roce 2009. Jeho výzkumné aktivity zahrnují optimalizace přenosu dat v přístupových širokopásmových sítích.

Jan VITÁSEK se narodil v Opavě. Titul Ing. získal na fakultě Elektrotechniky a komunikačních technologií v Brně studiem oboru Elektronika a sdělovací technika v roce 2009. Jeho výzkumné aktivity zahrnují šíření a zpracování optických signálů.