

Design of Colored Petri Net Simulation System Based on FPGA^{*}

CHEN Chengguan, ZHANG Xiaojun^{*}, ZHOU Taolue, ZHANG Dexue, GUO Hua

(Shandong University of Science and Technology, College of Electronic and Information Engineering, Qingdao Shandong 266590, China)

Abstract: In order to perform test of the large-scale colored Petri nets, a high-speed simulation system based on FPGA is designed. It implements the mapping of the place and transition modules to the hardware structures, and adopts C language to implement the hardware automatic generation tool for colored Petri nets. By analyzing the characters of the colored Petri nets, the tool generates the corresponding Verilog code and Quartus-based automatic script. Taking the model of “packet transmission” in communication as an instance, the generated code is tested in FPGA to verify its correctness.

Key words: Colored Petri Net; automatic generator tool; FPGA; Verilog HDL; simulation system

EEACC: 8520

doi: 10.3969/j.issn.1005-9490.2021.01.045

基于 FPGA 的有色 Petri 网仿真系统设计^{*}

陈成官, 张小军^{*}, 周韬略, 张德学, 郭 华

(山东科技大学电子信息工程学院, 山东 青岛 266590)

摘 要: 为了实现大规模有色 Petri 网模型的性能测试, 设计了基于 FPGA 的高速仿真系统。该系统实现了库所、变迁模块到硬件结构的映射, 采用 C 语言实现了有色 Petri 网的硬件自动生成工具。通过分析有色 Petri 网的特征, 该工具生成对应的 Verilog 代码和基于 Quartus 的自动脚本。以通信中“包传输”的模型为例, 在 FPGA 中对生成的代码进行测试, 验证了设计的正确性。

关键词: 有色 Petri 网; 自动生成工具; FPGA; Verilog HDL; 仿真系统

中图分类号: TP319

文献标识码: A

文章编号: 1005-9490(2021)01-0236-06

Petri 网 (Petri Net, PN) 是一种适合于描述异步、并发、分布式系统的抽象描述方式, 可用图形表示法和代数表示法分别表示, 具有直观性^[1]。由于这些特点, Petri 网的理论得以快速发展和应用, 其中有色 Petri 网 (Colored Petri Net, CPN) 已经被广泛用于系统仿真与验证^[2-4]、工作流分析^[5-7]、性能分析^[8-9]与算法错误检测^[10-11]等工作中。对于 PN 工具的设计与开发, 目前已经取得了许多研究成果^[12]。但是, 这些项目主要集中于通用计算机系统下对 P/T 网 (库所/变迁网) 进行的建模及模型的仿真验证; 并且在通用计算机系统上, 存在模拟 PN 并发性低效与仿真速度慢的问题。

现场可编程门阵列 (FPGA) 提供了一种新的建模方法。FPGA 与通用计算机系统相比具有速度快、可重构和能效比高的优势, 并且可直接描述

Petri 网异步并发的特性。随着 Petri 网规模的扩大与复杂度的增加, FPGA 的速度优势会越来越明显。目前对基于 FPGA 的 PN 仿真系统的研究较少, 且大多数涉及 Petri 网的研究, 都是以 Petri 网在实际问题的应用研究为导向的, 例如, Petri 网在逻辑控制器设计与离散事件系统监控理论中的应用^[13]; 并且现有的基于 FPGA 的 PN 实现方法不支持 CPN 的仿真^[14-15]。

本文实现基于 FPGA 的 CPN 仿真系统解决了两个问题: CPN 的硬件架构设计、CPN 硬件模型的自动生成。由于 CPN 的一般定义与描述特别抽象, 使得无法直接利用已有描述自动生成硬件模型, 因此本文设计了“CPN 描述文档”的内容及格式, 并实现了 CPN 硬件模型的自动生成工具, 该工具根据“CPN 描述文档”自动生成相应 CPN 模型与仿真控

项目来源: 山东省自然科学基金联合基金项目 (ZR2019ZLH001); 山东省重点研发计划项目 (2019GGX101066); 山东省高等学校青创科技计划项目 (2019KJN024, 2019KJN020)

收稿日期: 2020-03-16 **修改日期:** 2020-05-13

制器的 Verilog 代码描述。系统设计与验证方案如图 1 所示, 在已知 CPN 模型的基础上, 将 CPN 用“CPN 描述文档”进行描述, 通过自动生成工具生成 Verilog 代码描述; 经过 RTL 级仿真验证后, 建立 Quartus 工程, 经过编译, 在 Altera Cyclone V 5CSEMA5F31C6N FPGA 平台上运行测试。

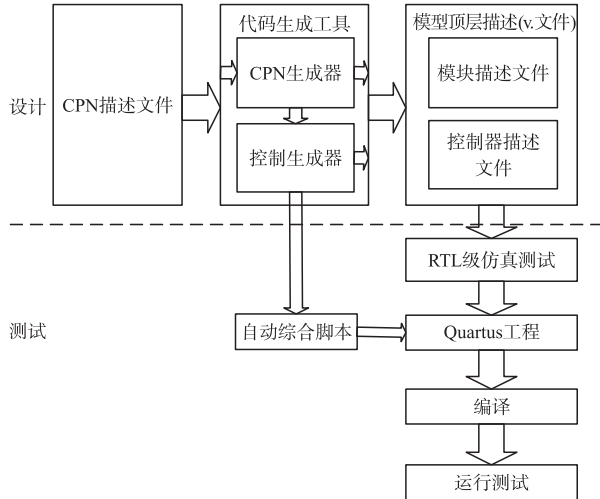


图 1 设计与验证方案

本文内容安排如下: 首先介绍了 CPN 的基本概念与定义, 并设计了 CPN 描述文件的内容格式与结构, 然后制定了 CPN 元素到 Verilog 描述的映射规则, 给出了自动生成工具的程序流程, 最后结合实例进行了运行测试。

1 有色 Petri 网与描述

1.1 定义

每个 PN 都由三部分组成: 表征状态的库所、表征动作的变迁以及表征流关系的有向弧。库所用于存储托肯(token); 流关系是权向量, 方向与有向弧相同, 其大小是权值, 变迁的发生通过权值对库所产生影响。在 CPN 中, 每个库所中可包含多种颜色的 token; 每个变迁可存在多种激活条件, 称为变迁的发生颜色; 每个有向弧可包含多个流关系, 每个流关系又对应变迁的一种或多种发生颜色。

一个经典 CPN 可定义为一个 9 元组 $N = \langle P, T, A, \Sigma, V, C, G, E, I \rangle$ [16-17]。其中:

- P 是库所的有穷集合, 在图形表示法中, 每个库所用一个圆形(○)表示;
- T 是变迁的有穷集合, 在图形表示法中, 每个变迁用一个矩形(□)表示;
- $A \subseteq P \times T \cup T \times P$ 是一个有限的“有向弧”集合, 用单向箭头(→)表示一个有向弧, Petri 网中用有向弧表征变迁与库所之间的流关系;

- Σ 是一个非空有限的颜色集的集合, 集合元素是颜色集;

- V 是变量的集合, $\forall v \in V, \text{Type}[v] \in \Sigma$, 变量的类型由颜色集限定, $\text{Type}[a]$ 表示 a 的类型;

- C 是库所的颜色集函数, 每个库所与 Σ 中的一个颜色集对应, 库所中 token 的颜色与颜色集中的元素一一对应;

- G 是从变迁 T 到变量表达式集 EXPR_V 的映射函数, 表征变迁的发生受到变量表达式的约束, $\forall \text{expression} \in \text{EXPR}_V, \text{Type}[\text{expression}] \in \Sigma$;

- E 是从弧 A 到变量表达式集 EXPR_V 的映射函数; 每个“有向弧”由一个变量表达式限定流关系, 通常, 变量表达式的所有取值对应该弧表示的所有流关系, $\forall a \in A, \text{Type}[E(a)] = C(p)$, 库所 p 与弧 a 相连;

- I 是 CPN 的初始化函数, 为 CPN 中每个库所分配一个初始化表达式及该表达式对应 token 的数量, $\forall p \in P, \text{Type}[I(p)] = C(p)$ 。

1.2 描述文档设计

CPN 与一般 P/T 网不同, 其复杂的流关系无法使用单一前后关联矩阵表示。为了描述 CPN 中的流关系, 本文提出了“库所的全局关联矩阵 W ”概念。为库所 P_n 定义一个描述其与变迁的流关系的矩阵:

$$W_n = N_{\text{CT-MAX}} \times N_T \quad (1)$$

式中: $n \in [1, N_p]$, N_p 是 CPN 中库所的数量; 列数 N_T 是变迁的数量; 行数 $N_{\text{CT-MAX}}$ 是单个变迁发生颜色数量的最大值。矩阵 W_n 中的每一项用常数“0”表示颜色 $C(p)$ 下变迁发生对库所 p 没有影响, 常数“1”表示颜色 $C(p)$ 下变迁发生会导致该库所中一种或几种颜色的 token 的数量增加, 常数“-1”表示颜色 $C(p)$ 下变迁发生会导致该库所中一种或几种颜色的 token 的数量减少, 常数“2”表示颜色 $C(p)$ 下变迁发生会导致使该库所中一种或几种颜色的 token 的数量增加的同时使该库所中 token 减少。

为增强 CPN 描述文档的可读性, 并清晰地描述变迁与库所间的流关系, 本文将库所的全局关联矩阵 W 拆成前向 W_{pre} 与后向 W_{post} 两个矩阵进行描述, 前向表征有向弧从变迁指向库所, 后向表征有向弧从库所指向变迁。将 W 中所有“-1”项置为 0, “2”项置为 1 得到 W_{pre} ; 将 W 中所有“1”项置为 0, “2”项置为 -1 得到 W_{post} 。描述文档包括 7 个部分:

(1) 库所与变迁的个数描述:

places_number(row) = N_p

transitions_number(column) = N_T

(2) 变迁发生颜色的个数描述:

[the number of transition's activation rules]

$N_{CT1} N_{CT2} \cdots N_{CTN_T}$

(3) 库所中 token 颜色的个数描述:

[color_number of places]

$N_{CP1} N_{CP2} \cdots N_{CPN_P}$

(4) Pre 描述:

定义表示流关系 (权向量) 的基本元素为字符串“颜色标识符:权值”,颜色标识符是流关系对应库所 token 的颜色,权值表示此流关系发生时对相应颜色 token 数量的影响;将 W_{pre} 的每一项用基本元素表示,不同项之间用空格隔开,同一项内的不同流关系之间用“+”连接;如, W_{pre} 的“0”项用“0:0”替换,“1”项用“sand:2+cement:1”替换;得到 $W_{s_{pre}}$ 描述格式如下:

[pre_matrix of petri net]

p1

$W_{s_{pre_of_p1}}$

p2

$W_{s_{pre_of_p2}}$

⋮

(5) Post 描述:

与(4)一致,先将 W_{post} 的每一项用基本元素表示得到 $W_{s_{post}}$ 。描述内容如下。

[post_matrix of petri net]

p1

$W_{s_{post_of_p1}}$

p2

$W_{s_{post_of_p2}}$

⋮

(6) 库所的 token 容限描述:

[the capacity of places]

p1

库所 1 中某颜色标识符 L

⋮

p2

库所 2 中某颜色标识符 L

⋮

(7) CPN 的 M_0 描述:

[initial token number of places]

p1

库所 1 中某颜色标识符 M

⋮

p2

库所 2 中某颜色标识符 M

⋮

上述描述中, N_P 是库所的个数, N_T 是变迁的个数; N_{CTi} 是变迁 i 发生颜色的个数; N_{CPj} 是库所 j 包含 token 的颜色的个数; L 是库所对该颜色 token 的最大容量; M 是在 CPN 的初始状态下库所中该颜色 token 的数量。

2 自动生成工具设计

2.1 CPN 的映射

本文将 CPN 状态的更新分成 3 步:变迁的预激活、变迁的激活与库所 token 的更新,每一步消耗一个时钟完成。变迁的预激活是根据库所中当前 token 数判断满足激活条件的变迁;在变迁的激活阶段对满足激活条件的变迁进行冲突的判断,并激活符合条件的变迁;在库所 token 的更新阶段,根据激活的变迁更新库所中 token 的数量,实现一次 CPN 状态的更新。

表 1 描述了从 CPN 描述到 Verilog 描述的映射规则。在 Verilog 设计中,将库所和变迁分别定义为一个 module 模块;将 token 定义为库所模块中的 output reg 类型的变量,位宽由库所的 token 容限描述确定;在 CPN 顶层模块中,为描述文档的 Pre/Post 描述中每个基本元素定义一个 wire,连接库所模块与变迁模块,并将权值定义为库所模块中的 parameter 参量。

表 1 映射

Token	库所	变迁	流关系的基本元素
output reg	module	module	wire
			权值 parameter

图 2 给出了 CPN 的 Verilog 模块的映射架构设计,共 4 种模块:顶层模块、控制器模块、库所模块、变迁模块;通过模块中的判断、判决与更新逻辑实现 CPN 的状态更新;reg 存储库所的 token 数。

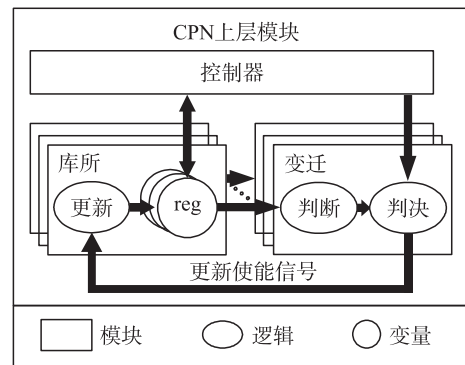


图 2 映射架构设计

2.2 工作流程

自动生成工具的工作流程如图 3 所示,主要包

括描述文件读取、参数分析、Verilog 描述文件打印和测试文件打印 4 个部分。测试文件包括用于 RTL 仿真的 testbench 文件和用于 Verilog 编译的自动综合脚本。参数分析过程会生成分析文件,可用于检验读取文件得到的 CPN 参数的正确性。

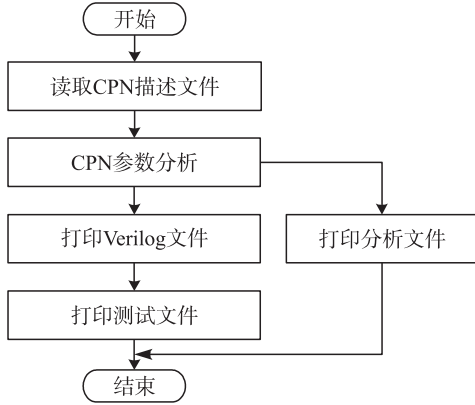


图 3 工作流程

3 测试

3.1 测试案例

本文选用一个描述通信中“包传输”的 CPN 模型^[18]作为测试案例,如图 4 所示,其颜色集和变量如表 2 所示。模型可以分为 3 个部分: Sender、Network 和 Receiver。库所 Send 存储要发送的字符串包及其编号,用颜色为(n,p)的 token 表示,如(1,“Modellin”)。变迁 SendPacket 的发生根据库所 NextSend 中字符串包的编号将字符串包发送到 Network 端的 A 库所。库所 E、F 与变迁 Ok 用来模拟网络 50% 的丢包率,本文假设丢包率为 50%。变迁 Update 根据库所 NextRec 中包的编号进行包的验证,若通过验证($n=k$),则将包中字符串与库所 Received 中原有字符串拼接得到的字符串更新到库所 Received 中,同时将 C 库所和 NextRec 库所中 token 颜色更新为下一个待传输包的编号。最后依次通过变迁 TransmitAck 和变迁 ReceiveAck 将下一个待传输包的编号更新到 NextSend 库所中,使变迁 SendPacket 开始发送一个新的字符串包。

表 2 “包传输”CPN 模型的颜色集和变量

```
ColorSet INT=integer;
ColorSet DATA=string;
ColorSet BOOL=bool;
ColorSet INT×DATA=product INT * DATA;
Variable n,k:INT;
Variable p,str:DATA;
Variable ok1:BOOL;
Variable ok2:BOOL;
```

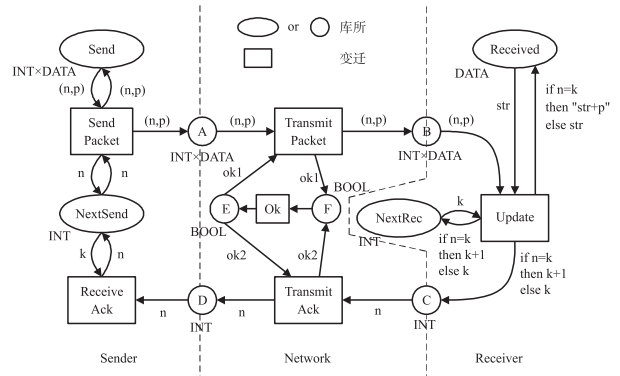


图 4 “包传输”CPN 模型

假设待传输包有 3 种,为(1,“Modellin”)、(2,“g and An”)和(3,“alysis”),分别记为包 1、包 2 和包 3,视为库所 Send 中 3 种不同颜色的 token,分别对应表 4 中 Send 行的颜色标识 C1、C2 和 C3;库所 E、F 包含 2 种 ok-token,分别用于变迁 TransmitPacket 与变迁 TransmitAck;将变量 n 和 k 的每种取值(1,2,3)都视作库所 NextSend、NextRec、C 和 D 中一种颜色的 token。本例中所有变迁的发生对 token 的数量消耗都是 1。本文根据该“包传输”CPN 模型得到表 3 所示的参数,并以此设计了描述文档。库所容限与 CPN 初始状态如表 4 所示。

表 3 “包传输”CPN 模型的参数

库所标识	token 颜色/个	变迁标识	发生颜色/个
Send(p1)	3	SendPacket(t1)	3
NextSend(p2)	3	TransmitPacket(t2)	3
A(p3)	3	Update(t3)	3
D(p4)	3	TransmitAck(t4)	3
E(p5)	2	ReceiveAck(t5)	3
F(p6)	2	Ok(t6)	2
B(p7)	3		
C(p8)	3		
NextRec(p9)	3		
Received(p10)	4		

为了简化表述,本文采用 p1、p2 等字符区分不同的库所,采用 t1、t2 区分变迁,采用 C1、C2 等字符作为 token 的颜色标识,区分库所中不同的 token,对应关系见表 3 与表 4。其中 p1、p3、p7 的颜色标识的含义对应相同,p2、p4、p8、p9 的颜色标识的含义对应相同,如 p1 的 C1 与 p3 的 C1 都指包 1,p2 的 C3 与 p8 的 C3 都指数字“3”。

表 4 CPN 的库所容限与初始状态

库所标识	颜色标识	初始 Token 数量	Token 容限
Send	C1	1	2
	C2	1	2
	C3	1	2
NextSend	C1	1	2
	C2	0	2
	C3	0	2
A	C1	0	1
	C2	0	1
	C3	0	1
D	C1	0	1
	C2	0	1
	C3	0	1
E	C1	1	1
	C2	1	1
F	C1	0	1
	C2	0	1
B	C1	0	1
	C2	0	1
	C3	0	1
C	C1	0	1
	C2	0	1
	C3	0	1
NextRec	C1	1	2
	C2	0	2
	C3	0	2
Received	C1	1	1
	C2	0	1
	C3	0	1
	C4	0	1

在 Received(p10)库所中,C1-token 是指空字符 token,即未接收到任何字符包时的 token,是 CPN 初

始状态下的 p10 库所包含的 token;C2-token 是接收包 1 后的 token;C3-token 是在 C2-token 的基础上接收包 2 后的 token;C4-token 是在 C3-token 的基础上接收包 3 后的 token。Received 库所中出现 C4-token 标志着 3 个包的成功发送。

3.2 测试结果

在 Altera Cyclone V 5CSEMA5F31C6N 平台上进行综合,综合频率为 156.37 MHz,ALMs 资源占用 781,Register 资源占用 2 027。CPN 状态每秒更新 5.2×10^7 次。并采用 Quartus 18.1 的 Signal tap 逻辑分析仪监控设计的运行,得到图 5 所示的信号波形图,图中 color2_of_place3 信号表示库所 p3 的 C2 颜色 token,其值是库所中该 token 的数量。

如图 5 所示,rst_n 信号置为高电平后(图中虚线),测试从表 4 第 3 列所示的 CPN 初始状态开始,此时库所 p1 中有 1 个 C1-token,库所 p2 中有 1 个 C1-token,满足变迁 t1 的发生条件,而其他变迁的发生条件不充足;因此从图中虚线后在 outclk_2 信号第 3 个时钟上升沿时只有 color1_of_place3 信号变高电平,而其他信号数值不变,表示变迁 t1 的发生在库所 p3 中产生了一个 C1-token 并且其他库所的 token 数量不发生变化。测试成功的标志,如图 5 中最下方 4 条信号波形所示,库所 p10(Received)的 4 种 token 从 C1-token 到 C4-token 依次出现,表示库所从空字符依次接收了 3 个字符包的过程,进一步对其他信号进行分析,信号的变化过程与 CPN 模型的规定一致,验证了自动生成工具的正确性。

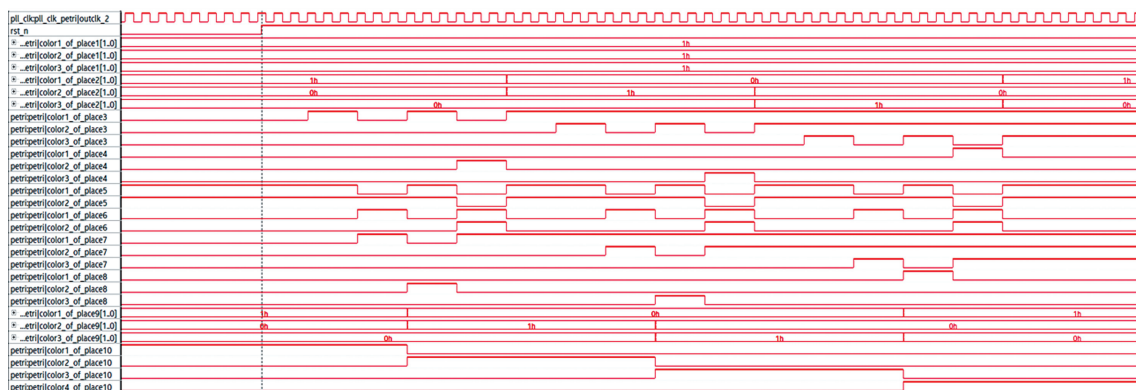


图 5 Signal tap 逻辑分析仪结果

4 结论

为了加速 CPN 模型的仿真,本文设计基于 FPGA 的系统进行加速。本文完成了 CPN 的模块到硬件结构的映射逻辑,设计了基于 C 语言的 CPN 模型的 Verilog 自动生成工具,实现了基于 FPGA 的

有色 Petri 网仿真系统设计。Signal tap 逻辑分析仪采集得到的信号验证了仿真系统的正确性。

参考文献:

- [1] 庞德强. Petri 网研究现状综述[J]. 现代交际,2016(22):144-145.
- [2] Vanit-Anunchai S. Modelling and Simulating a Thai Railway Sig-

- nalling System Using Coloured Petri Nets[J]. International Journal on Software Tools for Technology Transfer, 2018, 20(3): 243-262.
- [3] 钱文高,耿宏,马红岩. 装配序列约束下的飞机虚拟维修过程建模仿真[J]. 计算机工程与科学, 2019, 41(8): 1482-1489.
- [4] 惠馨雅,刘建华,刘浩. 基于有色 Petri 网的网络安全态势感知模型构建及分析[J]. 计算机与数字工程, 2019, 47(2): 393-396, 401.
- [5] Hu X, Jiao L. Efficient Modeling and Performance Analysis for IEEE 802.15.4 with Coloured Petri Nets[C]//2017 IEEE/ACM 25th International Symposium on Quality of Service(IWQoS), Vilanova de Geltru IEEE, 2017: 1-6.
- [6] Feng Z, Yuetong X, Jarong C. A Novel Petri Nets-Based Modeling Method for the Interaction Between the Sensor and the Geographic Environment in Emerging Sensor Networks[J]. Sensors, 2016, 16(10): 1-41.
- [7] Zairi S, Mezni A, Zouari B. Formal Approach for Modeling, Verification and Performance Analysis of Wireless Sensors Network [C]//International Conference on Wired/wireless Internet Communication, Malaga, Springer, 2015: 381-395.
- [8] 孙有朝,郭媛媛,崔灿丽. 应用有色 Petri 网的复杂系统四性一体化综合评估方法[J]. 南京航空航天大学学报, 2017, 49(5): 707-717.
- [9] Azgomi M A, Khalili A. Performance Evaluation of Sensor Medium Access Control Protocol Using Coloured Petri Nets[J]. Electronic Notes in Theoretical Computer Science, 2009, 242(2): 31-42.
- [10] 谢敏,吴亚雄,黄庶,刘明波. 基于有色 Petri 网的电网故障区域识别[J]. 电力系统保护与控制, 2016, 44(2): 56-64.
- [11] 李晨. 基于 Petri 网的地铁车辆远程故障诊断[J]. 微型电脑应用, 2020, 36(4): 50-53.
- [12] Petri Nets Tool Database[DB/OL]. Available: <http://www.informatik.uni-hamburg.de/TGI/PetriNets/tools/db.html>.
- [13] 罗继亮,邵辉,吴维敏,苏宏业. 逻辑控制器设计与离散事件系统监控理论[J]. 控制理论与应用, 2018, 35(1): 86-91.
- [14] Silva C F, Quintáns C, Colmenar A, et al. A Method Based on Petri Nets and a Matrix Model to Implement Reconfigurable Logic Controllers[J]. IEEE Transactions on Industrial Electronics, 2010, 57(10): 3544-3556.
- [15] Gomes L, Costa A, Barros J P, et al. From Petri Net Models to VHDL Implementation of Digital Controllers [C]//33rd Annual Conference of the IEEE Industrial Electronics Society, Taipei, China, IEEE, 2007: 94-99.
- [16] Wu X Y, Wu X Y. Extended Object-Oriented Petri Net Model for Mission Reliability Simulation of Repairable PMS with Common Cause Failures[J]. Reliability Engineering and System Safety, 2015, 136: 109-119.
- [17] Zhao J T, Chen Z Q, Liu Z X. A Novel Matrix Approach for the Stability and Stabilization Analysis of Colored Petri Nets[J]. Science China(Information Sciences), 2019, 62(9): 102-115.
- [18] Jensen K, Kristensen L M. Coloured Petri Nets: Modelling and Validation of Concurrent Systems[M]. Springer-Verlag Berlin Heidelberg, 2009: 13-41.



陈成官(1998—),男,汉族,山东巨野,硕士研究生。研究方向为数字 IC 设计,ward_chen1998@163.com;



张小军(1980—),男,汉族,山东博兴,副教授,硕士生导师。研究方向为数字 IC 设计、信道编解码等,buttern@163.com。