doi:10.3969/j.issn.1671-637X.2013.11.017

时间自动机流量特性的硬件模拟

陈 亚, 李 峭, 赵露茜
(北京航空航天大学,北京 100191)

摘 要: 在实时网络中,受到突发度等指标约束的流量特性是最坏情况下服务质量保证的关键因素之一。采用时间自动机对流量进行模拟,可以反映在到达曲线的组合约束下流量的不确定性。采用硬件描述语言对相应的时间自动机形式化模型进行转换,研究了硬件逻辑与时间自动机模型的对应方法,利用可编程阵列芯片并发运行的优势,充分体现了时间自动机之间的并发行为,用于进行硬件在回路测试。采用此方法构成流量特性模型的转换接口,并采用硬件描述语言实现,经过在典型测试用例下的仿真测试,发现该装置能够根据模型参数对于虚拟链路的流量特性进行约束,模拟生成综合化网络中的实时通信流量,说明了该硬件模拟方法的可行性。

关键词: 实时系统; 时间自动机; 流量特性; 硬件在回路
中图分类号: V243; TP393 **文献标志码**: A **文章编号**: 1671-637X(2013)11-0078-06

A Hardware Implementation to Emulate Traffic Characteristics Based on Timed Automata Models

CHEN Ya, LI Qiao, ZHAO Luxi
(Beihang University, Beijing 100191, China)

Abstract: In real-time networks, traffic characteristic with burstiness constraints is one of the key QoS (Quality of Service) factors in the worst case. Using timed automata to model the traffic can reflect the uncertainty under the constraints by arrival curves. We adopted hardware description language to describe timed automata formal models, and studied correspondence between hardware logic and timed automata model. The method took the advantages of programmable array chips, fully represented the concurrent behaviors between timed automata, and then implemented the hardware-in-the-loop testing and detection. With such a method, we simulated the virtual link traffic generation in real-time network. Some typical test results indicated that the model can constrain the traffic characteristics of virtual link, which proves the feasibility of the hardware implementation method.

Key words: real-time system; timed automata; traffic characteristic; hardware-in-the-loop

0 引言

时间自动机[1](Timed Automata,TA)理论是一种用于实时系统模型检查(model checking)的形式化建模与验证方法[2]。时间自动机模型由位置、转移条件、时钟变量和同步信道等建模要素构成,最初被用于基于软件遍历搜索的模型检查,而它们的扩展应用包括利用时间自动机生成测试用例(UPPAAL-CoVer)[3],以及采用PC机上运行的模型检查引擎进行在线测试(UPPAAL-TRON)[4],也有学者提出将时间自动机模型与可编程硬件相结合,实现硬件在回路(Hardware-In-the-Loop,HIL)运行[5],但依赖于采用Xilinx System Generator工具将Matlab代码转换为FPGA硬件电路。

对于嵌入式实时通信网络,如果是事件驱动的通信,可以采用实时演算(Real-Time Calculus,RTC)与TA模型相结合的方法[6]对分布互连的嵌入式实时系统进行分析,其中需要利用TA模型对网络中的流量特性进行模拟,即对并发的虚拟链路流量进行突发度的约束,形成对系统的事件序列激励。TA模型便于描述并发实时性能以及对流量特性进行建模和对其特性进行模拟。本文提出基于时间自动机模型的流量特性的硬件模拟

收稿日期:2012-12-06 **修回日期**:2012-12-31
基金项目:国家自然科学基金(61073012);国家高技术发展研究计划("八六三"计划)(2011AA110101)
作者简介:陈 亚(1990—),女,河南信阳人,硕士生,研究方向为实时通信系统的测试验证方法。

方法，与文献[5]中使用顺序执行的编程方法不同，直接使用硬件描述语言，明确了在流量特性模拟过程中用硬件描述语言编写的模块与时间自动机模型之间的对应关系，以充分利用硬件资源的并行执行特性。

与软件仿真验证相比，本文所述的基于时间自动机的流量特性的硬件模拟方法将时间自动机模型直接与硬件描述语言对应，运行速度原理上仅受到芯片内部时钟和硬件连线速度的制约，在保证线速并发处理的同时，满足实时系统对时间准确性的要求。

1　模型及其转换原理

时间自动机模型由位置、有向边、时钟变量、同步信道变量、本地变量和全局变量等要素构成[1]，对应位置有不变量约束，对应有向边有守卫条件、同步信道耦合和时钟更新。当有向边上的守卫条件和目的位置上的不变量条件同时被满足时，可以执行位置的转移。

对系统的描述需要用一组时间自动机模型，多个时间自动机之间的交互是利用全局变量和同步信道实现的。其中，同步信道包括二元同步信道和广播信道[7]，在硬件实现中，各个自动机实例被封装为不同的模块，模块之间的通信则与 TA 模型的全局变量和信道的同步操作有关，是 TA 模型与硬件逻辑对应的关键。

对于分布互连的嵌入式实时系统的分析，可以将基于 RTC 的分析方法和基于 TA 模型的形式化方法相结合，形成一种混合式的分析方法[6]。本文对于实时通信流量的模拟即引进了这种方法，以避免仅使用 TA 模型的形式化方法可能引起的内部状态空间爆炸问题。RTC 模型给出流量的到达曲线约束参数，通过转换接口将一系列的到达曲线约束转换为 TA 模型，本文也采用硬件逻辑实现了该转换接口。

对抽象的流量模型（如到达曲线）进行模拟，产生能够驱动系统 TA 模型的事件序列激励。以 R^{α} 和 R^{TA} 分别表示 RTC 和 TA 模型下的流量约束（具体解释见 3.1 节），将转换接口记为 T，图 1 为系统的 RTC 模型与 TA 模型之间转换接口的示意图。

模型转换条件成立的条件为：当且仅当 $R^{\mathrm{TA}} \supseteq R^{\alpha}$，TA 模型所描述的流量模型才是正确的，即

$$r \models \alpha \Rightarrow r \in R^{\mathrm{TA}}, \forall r \in \mathbf{R} \qquad (1)$$

其中，R 表示所有可能的事件轨迹。

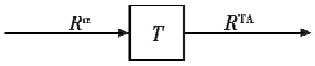

图 1　RTC 模型到 TA 模型的转换接口

Fig. 1　Interface from RTC to TA

2　时间自动机模型与硬件逻辑的对应关系

事件同步是时间自动机区别于有限状态机的重要特点之一，是硬件实现的关键问题。不同于 TA 模型中由符号演算关系表达的抽象时钟变量，实现中的硬件时钟在离散时间点不断累加计时，这既与抽象时钟变量有区别，也需要与真实的物理时间推进相协调，如何使之既正确反映 TA 模型的语义，又与真实的物理时间的差别在可接受的容忍范围之内，即合理地在硬件上实现时间自动机的时钟，也是亟待解决的关键问题。另外，时间自动机模型的实例之间的相互协调也需要全局变量的支持。

2.1　两个时间自动机的同步

最基本的事件同步涉及两个时间自动机，如图 2 所示，TA_1 和 TA_2 开始都分别处于初始状态 S_1，当 TA_1 释放 event 事件（用 event！表示），TA_2 就用 event？与 TA_1 进行同步，两个 S_1 状态的下一条边同时被触发。一对“！”和“？”组成一组同步事件，称为二元同步信道。

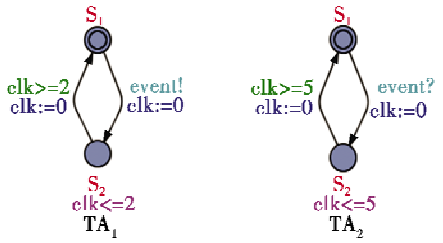

图 2　两个时间自动机的事件同步

Fig. 2　Binary synchronization between two TAs

二元同步信道的硬件实现原理图如图 3 所示，TA 模型中通过同步信道变量“？”和“！”耦合的二元同步操作需要在硬件中以真实信号同时使能及其确认的握手操作实现。在图 3 给出的例子中，设定两个时间自动机 TA_1 和 TA_2 的硬件实现分别有两个变量：事件请求变量 $event\_req\_a_i$ 和确认变量 $event\_ack\_a_i$（$i=1,2$），$event\_req\_a_i$ 用于启动同步过程，$event\_ack\_a_i$ 返回同步确认信息。同步信道对请求同步信号进行“与”操作，经过寄存器返回确认信号。即：TA_1 和 TA_2 同时请求 event 事件时，$event\_req\_a_i$ 为 1，经过与操作和寄存器返回的 $event\_ack\_a_i$ 也是 1，说明 TA_1 和 TA_2 的event事件同步成功。

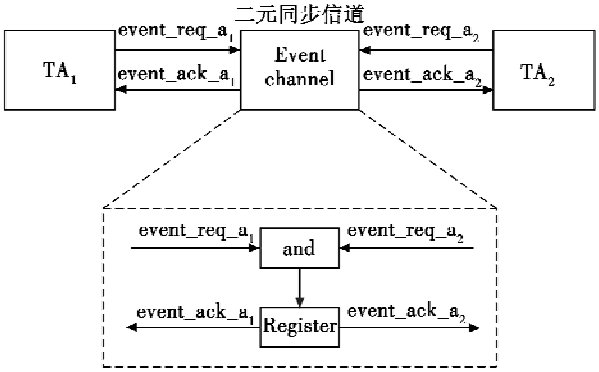

图 3　二元同步信道的硬件实现

Fig. 3　Hardware implementation of binary synchronized channel

2.2 多个时间自动机的同步

图4为多个时间自动机共用广播信道的示意图，其中，依赖控制器 controller 的 channel! 信号与 TA_1、TA_2 和 TA_3 中的一个或多个 channel? 信号同步来驱动事件的发生。

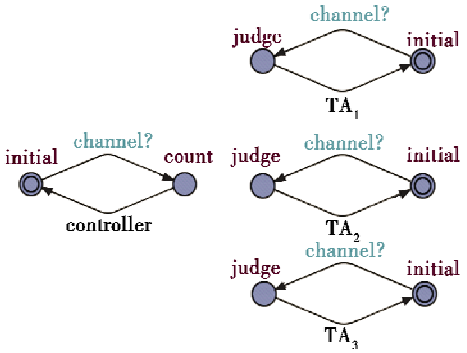

图4 多个时间自动机的事件同步示意图

Fig. 4 Broadcast synchronization among several TAs

图5为多个时间自动机之间同步的硬件实现，当控制器 controller 处于 initial 状态时，发出请求同步的信号 channel_req_c，此时，只要 TA_1、TA_2 和 TA_3 其中至少有一个时间自动机也发出请求同步的信号，控制器和请求同步的时间自动机的状态就同时从初始状态跳转到下一个状态，即图4中 channel 边的跳转被激活。像这样多个 TA 模型之间的同步信道称为广播信道。

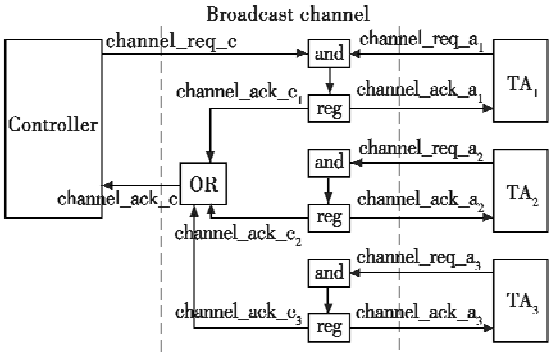

图5 广播同步信道的硬件实现

Fig. 5 Hardware implementation of broadcast channel

2.3 时间自动机时钟的表示

在经过了硬件扩展的模型检查实验中，时间自动机的时钟与现实的物理时间有一定关系——以一定单位的离散时间计时，如果这种离散化在合理的范围内，它使自动机的运行能够模拟真实系统的行为，以便对运行中的实时系统进行在线测试和监测。

本文对时钟的处理方法如图6所示。其中：osc 为 FPGA 的主频；timer 为时间自动机的时钟；flag 为由时间自动机发送给时钟模块的置零标志。物理时钟经过分频并计数为时间自动机的模型实现提供时钟，时钟发生器是与时间自动机并行运行的硬件模块，当自动机需要对时钟进行复位操作时，就发出置零标志，时钟发生器模块就对计数器进行复位。

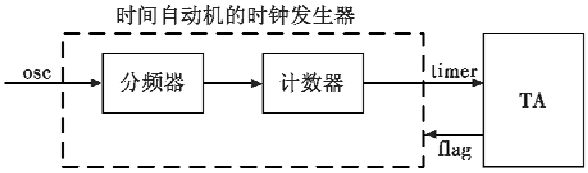

图6 时间自动机时钟的实现

Fig. 6 Implementation of clocks in TA

2.4 全局变量的实现

通过多个时间自动机之间的协调运行能够模拟实时系统的并发运行的组件，多个自动机的协调除了需要同步信道以外，还要用到的是全局变量。例如，图7展示的是以 sync 变量作为自动机 controller、u_1、u_2 和 u_3 共用的全局变量，实现的功能是：controller 发出事件 event!，与此同时，当 u_1、u_2 或 u_3 请求事件 event? 时，其各自分别使 sync 加1，当全局变量 sync 等于某一阈值（本例中等于3），controller 自动机的位置才能从 initial 跳转到 count。注意自动机 controller 的 count 位置被设置成紧急（urgent）类型，表示在 count 位置没有时间停留，意味着在此之前如果 u_1、u_2、u_3 不是同在 initial 位置，则 controller 在 count 位置将会返回不变量关系式不被满足的异常。

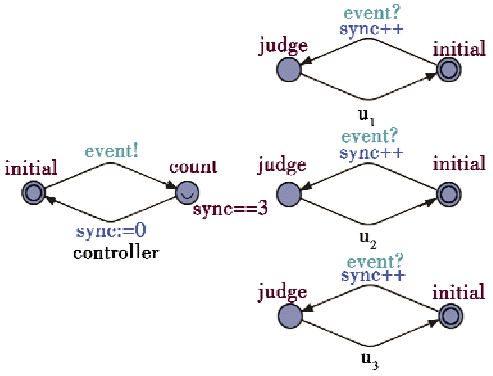

图7 时间自动机的全局变量示意图

Fig. 7 Global variable in TA

对于全局变量的硬件实现，采用一个加法器和置零的反馈线。图8所示的原理图就是图7中自动机实例的硬件实现，u_1、u_2 和 u_3 的 sync 变量分别用 $sync_1$、$sync_2$ 和 $sync_3$ 代替，将它们3个的和传递给 controller 模块，同时 controller 的 sync 变量置零的时候，也要通知 u_1、u_2 和 u_3 模块置零 $sync_1$、$sync_2$ 和 $sync_3$ 变量。

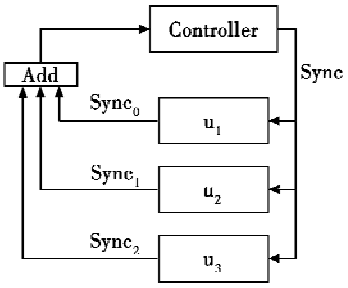

图 8　全局变量的硬件实现

Fig. 8　Hardware implementation of global variable

3　流量特性模型的转换接口

对于到达曲线 α 描述的虚拟链路流量特性，硬件在回路测试中的关键问题在于：对于给定的到达曲线，要将满足该模型的流量转换成完全对应的到达事件轨迹，而且要模拟在此约束下数据包到达事件的不确定性。在任意时间间隔 Δ 内，满足到达曲线 α 的可能事件轨迹是一个无穷集合，采用 TA 模型恰好能描述这种不确定性。

对流量的累积函数 $R(t)$ 进行定义[8]，使它表示在 $[0,t]$ 时间内在流量上传输的比特数。在流量模型中，将 $R(t)$ 考虑成事件（数据包）的到达轨迹。

对于离散化的到达曲线约束，可以将到达曲线 α 用一组标准阶梯函数（如图 9 所示）α_i 来逼近，即取所有标准阶梯函数的最小值来逼近

$$\alpha = \min_i(\alpha_i) \qquad (2)$$

并设：通过转换接口 T 的设计，可以如图 1 及其充要条件所示，保证 TA 模型的参数设定正确表达相应的到达曲线约束。

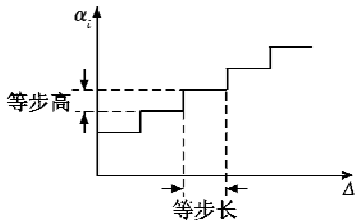

图 9　标准阶梯函数示意图

Fig. 9　Standard staircase function

每个阶梯函数 α_i 分别对应一个时间自动机 TA_i 的实例，它们通过相互之间的同步信道相互制约，产生数据包，得到事件轨迹（见图 10）：

$$r = ((packet,t_0);(packet,t_1);\cdots;(packet,t_n)) \qquad (3)$$

本文用 3 条阶梯函数曲线 α_1、α_2 和 α_3 共同约束，来逼近某条具有多段不同阶梯间隔和高度的到达曲线 α（图 11 中实线所示）。其中，N_1、N_2 和 N_3 分别是 3 条阶梯函数曲线的突发容量，表示零时刻产生的事件个数，之后在每个上升沿时刻产生一个事件。δ_1、δ_2 和 δ_3 表示了 α 的两个连续事件的最小间隔。

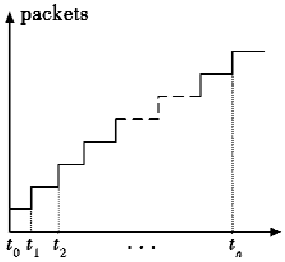

图 10　事件轨迹

Fig. 10　Event trace

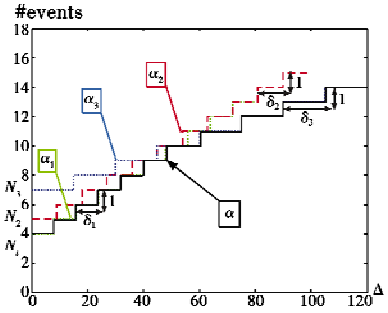

图 11　三条阶梯函数逼近的到达曲线 α

Fig. 11　Approximation of the arrival curve α using three staircase functions

图 12 为转换接口的 TA 模型示意图。

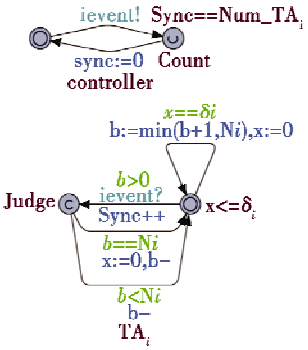

图 12　转换接口的时间自动机模型（$i=1,2,3$）

Fig. 12　TA models of the interface ($i=1,2,3$)

整体来看，图 1 所示的转换接口接收原始的事件到达轨迹的激励，将用到达曲线描述的事件轨迹转换到 TA 模型描述的网络模型中，并根据 TA 模型在线判断它们是否符合给定的流量约束，对于不符合要求的事件序列进行抑制。它的用途在于生成完全符合流量约束的到达流量，作为网络中的流量输入。

TA_i（$i=1,2,3$）分别是到达曲线 α_i 的时间自动机模型，每个 TA 模型都有各自的时钟变量 x、突发度变量 b 和零时刻的突发度常数 N_i 及阶梯间隔 δ_i。控制器 controller 通过事件 ievent 和全局变量 Sync 控制 TA_1、TA_2 和 TA_3 之间的状态转移时机，并在适当的时候复位时钟 x。这一组时间自动机模型相互协作，生

成完全符合到达曲线 α 约束的到达事件轨迹，作为网络中的输入。

4 转换接口的硬件模拟

在实现了转换接口的条件下，只需用没有经过整形的数据包到达事件馈入转换接口，就可以模拟生成符合到达曲线约束设定的虚拟链路流量，其原理如图13所示。当然，也可以将已经满足某种流量特性约束的数据包到达事件馈入，实现流量管制意义下的多重流量特性约束的实验。

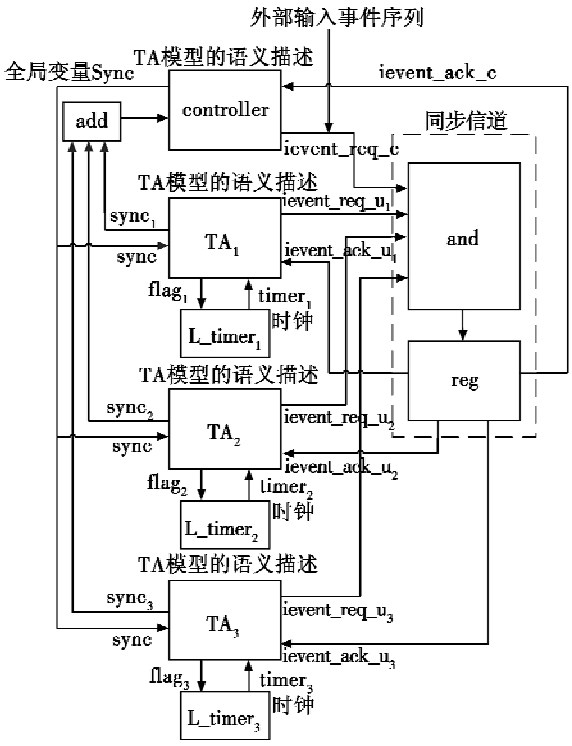

图13 转换接口的硬件模拟图

Fig. 13 Hardware simulation of the interface

本文利用 Xilinx Virtex 4 XC4VFX12-F668-10C 芯片，根据第2节所述的 TA 模型到硬件逻辑的转换方法，在 Xilinx ISE 环境下将转换接口的 TA 模型用硬件逻辑进行仿真模拟。该模型主要分为4个部分，分别是 TA 模型的语义描述，时钟，同步信道和全局变量。语义描述模块是对控制器(controller)和到达曲线的 TA 模型($TA_i(i=1,2,3)$)的语义描述。本地时钟模块 $L\_timer_i(i=1,2,3)$ 分别为时间自动机 TA_i 提供计数时钟。这里利用广播信道，实现各个时间自动机模型之间 event! 和 event? 事件的同步。全局变量 Sync 是由一个加法器(add)和控制器到 TA_i 的馈线实现。设计该转换接口主要用于将流量特性的 RTC 模型转换为满足约束条件的事件序列，产生能够驱动 TA 描述的网络模型的流量输入，如图13所示。

5 转换接口的功能测试分析

针对硬件模拟的流量特性模型的转换接口，本文设计了3种典型的测试用例，如图14所示，即 r_1、r_2 和 r_3 3种事件轨迹，分别是低于 α 约束、部分恰好达到 α 约束和部分超过 α 约束的情况，将它们分别作为流量模型的转换接口的原始激励事件序列，考察前者是否根据到达曲线对它们进行合理的抑制。

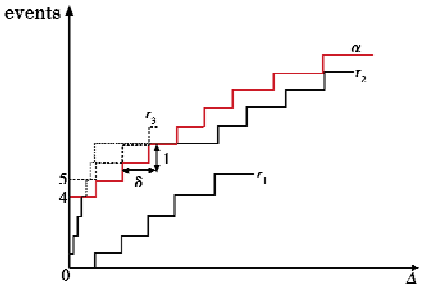

图14 事件轨迹的3种情况

Fig. 14 Three cases of input curves

本文在 Xilinx ISE 的仿真环境下，将3种输入事件序列作为转换接口模型的输入，观测转换接口的输出事件序列，其中，采用脉冲信号模拟事件的到达(在真实系统中，这些脉冲将对应物理层接口实际收到数据包)，并将突发容量等效为间隔很近的几个有序脉冲。

通过对比转换接口的输入事件序列与输出事件序列的时序仿真波形就可以看出，该系统模拟生成了符合流量特性约束的事件到达轨迹，并对不符合流量特性约束的事件轨迹进行转换。

当输入符合 r_1 轨迹的事件序列时，输出事件序列应与输入一致，如图15所示。

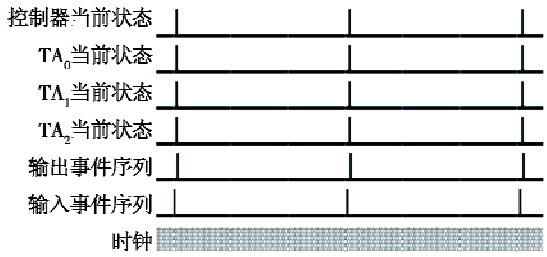

图15 输入事件符合 r_1 的测试波形

Fig. 15 Test waveform when the input curves is r_1

当输入符合 r_2 轨迹的事件序列时，事件轨迹超出到达曲线 α 的部分(如图14所示 r_2 的虚线部分)将会被约束在到达曲线 α 上，在虚线部分对应的输入事件将不会被响应，如图16所示。

另一种较为特殊的情况是，当输入事件轨迹在零时刻的突发容量就大于到达曲线的突发容量时，也能出现输出与输入的不一致，说明受到了转换接口的抑制，如图17所示。

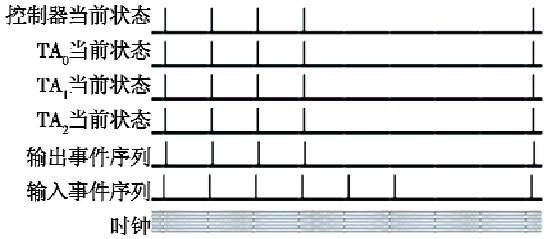

图 16　输入事件符合 r_2 的测试波形

Fig. 16　Test waveform when the input curves is r_2

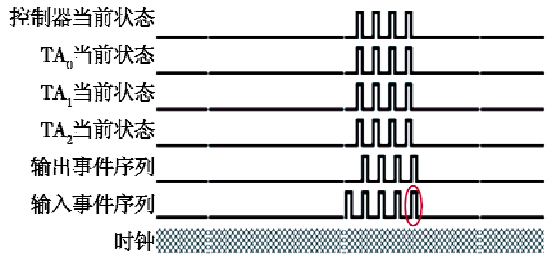

图 17　输入事件符合 r_3 的测试波形

Fig. 17　Test waveform when the input curves is r_3

6　结束语

本文所述的工作立足于时间自动机的硬件实现，在线模拟网络节点流量特性并进行必要的流量整形，给出了时间自动机模型与硬件逻辑的对应方法，设计实现了一种能够衔接 RTC 分析模型和 TA 模型的流量特性硬件转换接口，并通过功能仿真验证了其对于流量特性约束和模拟的正确性。这些工作表明并验证了用硬件实现时间自动机模型的可行性，为进一步验证分布式实时系统的多组件的协议行为和在线监测研究打下基础。

参 考 文 献

[1] RAJEEV A, DAVID L D. A theory of timed automata [J]. Theoretical Computer Science, 1994, 126(2):183-235.

[2] 孙全勇. 时间自动机及其应用研究[D]. 哈尔滨：哈尔滨工程大学，2006.

[3] ANDERS H, LARSEN K G, MIKUCIONIS M, et al. Testing real-time systems using UPPAAL[M]. Formal Methods and Testing, LNCS 4949, Berlin Heidelberg: Springer-Verlag, 2008:77-117.

[4] LARSEN K G, MIKUCIONIS M, NIELSEN B, et al. Testing real-time embedded software using UPPAAL-TRON: An industrial case study [C]//ACM. EMSOFT'05, Jersey City, New Jersey, USA, 2005:299-306.

[5] KROKORA J, HANZALEK Z. FPGA based tester tool for hybird real-time systems [J]. Microprocessors and Microsystems, 2008, 32(8):447-459.

[6] LAMPKA K, PERATHONER S, THIELE L. Analytic real-time analysis and timed automata: A hybrid methodology for the performance analysis of embedded real-time systems [J]. Design Automation for Embedded Systems, 2010, 14(3):193-227.

[7] BEHRMANN G, DAVID A, LARSEN K A. A tutorial on UPPAAL [Z]. Denmark: Department of Computer Science, Aalborg University, 2004.

[8] LE BOUDEC J Y, THIRAN P. Network calculus: A theory of deterministic queuing systems for the internet [M]. New York: Springer-Verlag, 2001.

(上接第 77 页)

[8] HADRI A E, BENALLEGUE A. Attitude estimation with gyros-bias compensation using low-cost sensors[C]//Joint 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference Shanghai, P. R. China, December 16-18, 2009:8077-8082.

[9] KUMAR N S, JANN T. Estimation of attitudes from a low-cost miniaturized inertial platform using Kalman filter-based sensor fusion algorithm[R]. Sadhana, 2004.

[10] 崔亚琦，熊伟，何友. 基于 MLR 的机动平台传感器误差配准算法[J]. 航空学报，2012，33(1):118-128.

[11] HELMICK R E, CONTE H E, RICE T R. Absolute sensor alignment using GPS[C]//SPIE, 2739:168-179.

[12] WATSON G A, RICE T R. Sensor alignment and compensation for composite tracking [C]//Proceedings of SPIE 4728, 2002:354-367.

[13] HERMAN S M, POORE A B. Nonlinear least-squares estimation for sensor and navigation biaes[C]//Proceedings of SPIE, 2006, 6236:623617-1-623617-18.

[14] 潘江怀，何佳洲，罗双喜. 舰载雷达探测误差传递与灵敏度分析[C]//第三届中国信息融合大会论文集，西安，2011:514-520.