软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2016,27(Suppl.(2)):50−57

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

一种基于状态机的硬件分组处理监测技术*

厉俊男, 胡 锴, 李 韬, 唐 路

(国防科学技术大学 计算机学院,湖南 长沙 410073)

通讯作者: 厉俊男, E-mail: nudt_ljn@163.com

摘 要: 网络设备硬件的封闭性使科研及教学人员难以获知网络设备硬件的实现细节,网络设备硬件的科研和教学面临巨大的挑战.针对上述问题,提出了一种基于状态机的硬件分组处理监测技术 PktScope,通过设计硬件分组处理状态机编码规范并制定标准的分组数据及流程上报接口,实现对硬件进行分组处理过程的实时监测,可有效应用于网络设备的科研及教学过程.基于 MagicRouter 的原型系统实验结果表明,PktScope 技术可以有效地实现对网络流量的流级实时监测,并且具有硬件资源额外开销低的特点.

关键词: 网络;分组处理;态监测;状态机

中文引用格式: 厉俊男,胡锴,李韬,唐路.一种基于状态机的硬件分组处理监测技术.软件学报,2016,27(Suppl.(2)):50−57. http://www.jos.org.cn/1000-9825/16018.htm

英文引用格式: Li JN, Hu K, Li T, Tang L. A FSM-based hardware monitoring technology for packet processing. Ruan Jian Xue Bao/Journal of Software, 2016,27(Suppl.(2)):50−57 (in Chinese). http://www.jos.org.cn/1000-9825/16018.htm

A FSM-Based Hardware Monitoring Technology for Packet Processing

LI Jun-Nan, HU Kai, LI Tao, TANG Lu

(College of Computer, National University of Defense Technology, Changsha 410073, China)

Abstract: The closure of network equipment hardware makes scientific research and teaching staffs difficult to learn the details of the network device hardware. Hardware research and teaching face enormous challenges. To solve these problems, this paper proposes a hardware packet processing monitoring technology based on Finite State Machine called PktScope, to monitor the processing of packets in the hardware in real time by designing hardware packet processing state machine coding standard and formulating standard packet data and processing reporting interface, which can be effectively applied to the network equipment research and teaching field. The prototype system is implemented based on MagicRouter. The PktScope technology can effectively realize the real-time monitoring of network flows, and has the characteristic of low additional hardware resource consumption.

Key words: network; packet processing; state monitoring; FSM

作为网络底层硬件的路由器、交换机等网络设备,随着对其功能需求的增大,网络设备的研制要求也不断提升.然而,网络设备硬件就是一个“黑盒子”,使科研及教学人员对其内部的分组处理过程无法直接观察,给网络科研创新及教学活动带来巨大挑战.为了实现对设备内部数据处理过程的监测,现有研究通常利用一些软硬件的平台,来获取需要的数据处理信息.但这些平台通常难以动态获取硬件分组相关数据及内部状态变化信息,少数平台虽然提供了硬件信号级别监测,但其操作的复杂性和专业性给掌握硬件处理数据过程带来很大的困难,且未提供易于理解的抽象视图.

针对上述问题,本文提出一种基于状态机的硬件分组处理监测技术——PktScope 技术.首先,该技术基于状

* 基金项目: 国家高技术研究发展计划(863)(2015AA010201); 国家自然科学基金(61202483, 61202485)

Foundation item: National High-Tech R&D Program of China (863) (2015AA010201); National Natural Science Foundation of China (61202483, 61202485)

收稿时间: 2015-05-31; 采用时间: 2016-01-05

态机(又称有限状态机,finite-state machine,简称 FSM),状态机是表示有限个状态以及在这些状态之间的转移和动作等行为的数学模型.通过状态之间的转移和动作清晰完整地展示硬件分组处理的过程.其次,该技术可以对分组进行实时流级监测,通过在线实时上传硬件分组处理数据及状态信息,可进行系统抽象展示.在网络科研方面,PktScope 可有效支持科研的原型系统研制,对整个原型系统进行监测,及时直观地向科研人员提供原型系统数据处理过程信息,利于硬件功能调试,并支持通过对硬件分组处理状态的掌握,加快原型系统研制进度.在网络教学方面,PktScope 可实时反映监控流的处理信息,增强网络教学的直观性,加深学生对网络设备硬件处理数据过程的理解.

本文第 1 节对传统的监测技术及其不足进行讨论.第 2 节对 PktScope 技术的设计思想和关键技术进行详细介绍.第 3 节介绍 PktScope 技术的实验验证.最后是对全文的总结和展望.

1 相关研究

1.1 SDB 的基本思想

传统的监测技术大多基于网络教学科研平台提供,通常可分为两类,一类是基于软件的网络仿真模拟平台,另一类是基于硬件开发平台的信号捕获工具.

(1) 基于软件的网络仿真平台

NS2(network simulator version 2)[1]是一种面向对象的网络仿真器.通过网络仿真,能对各组件的行为进行较精确的模拟,获得足够数据对系统的性能进行较准确的预测,利用和整合现有的网络资源,使网络达到最高性能.然而 NS2 主要面向网络层及以上层,可以看到路由器之间的处理数据过程,缺乏对网络底层硬件内部处理数据的关注.

Click[2]是一种新型模块化的软件路由器体系结构,它采用面向对象的模块化设计方式,根据用户的需要来选择组成模块定义路由器功能,使路由器软件更加灵活且易于配置和管理.通过不同功能模块的组合体会路由器实现功能的不同.Click 可以反映硬件的分组处理过程,但其是基于软件仿真方式实现.

上述技术大多基于软件仿真模拟,难以实时提供网络底层硬件真实的分组处理信息.

(2) 基于硬件开发平台的信号监测工具

Signaltap[3]是一款 Altera 针对其 FPGA 提供的系统级信号调试工具,可以选择要捕获的信号特征、捕获时间,以及数据样本大小,将实时数据以波形图的方式提供分析,得到分组处理过程的信息.但 Signaltap 依赖工程中剩余的 RAM 资源.若资源不足,无法进行大量分组数据及处理状态的连续采集展示.

Chipscope[4]与 Signaltap 类似,是 Xilinx 公司针对其 FPGA 提供的一款在线调试软件,也是一个逻辑分析仪,主要用于在上板测试过程中采集并观察芯片的内部信号,监控硬件处理过程,以便于调试.

上述平台监测技术专业性强,复杂度高.并且,受限于硬件内部资源空间,难以进行全流程系统级监测.此外,提取的波形图缺乏直观性.

针对上述问题,本文提出了一种基于状态机的硬件分组处理监测技术 PktScope,通过制定硬件分组处理状态机编码规范并设计标准的分组数据及流程上报接口,实现对硬件进行分组处理过程进行实时监测.

2 PktScope 技术

2.1 设计思想

PktScope 的设计原则是基于状态机的描述方式,尽量避免干涉或约束待监测的硬件模块的分组处理功能实现,获取更为详细的硬件分组处理过程数据及状态信息,实现对硬件分组处理的监测.

如图 1 所示,PktScope 的基本原理如下:

(1) 制定待检测硬件逻辑模块的状态机编码规范,并设计标准的分组数据及流程上报接口,以供监测部件

进行硬件分组处理信息的收集和上报.

(2) 基于上述的编码规范和接口,根据由软件下发的分组的流信息(源 MAC 地址,目的 MAC 地址,源 IP 地址,目的 IP 地址,传输协议类型号),锁定监测对象流.对收集的分组数据和状态信息进行筛选、封装、上报.

(3) 由软件配置模块下发监测规则(分组的流信息),并由分析模块基于上报的分组数据及流程进行分析汇总,形成监测结果展示模块将监测结果实现系统级可视化展示.

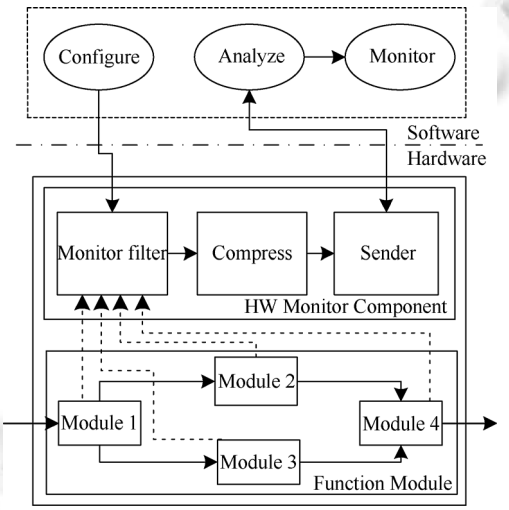

Fig.1 System architecture of PktScope

图 1 PktScope 系统架构

2.2 关键技术

PktScope 涉及的关键技术主要包括待监测模块的设计规范、监测报文格式定义及软件分析技术.

2.2.1 待监测模块设计规范

为保证硬件分组处理数据及状态信息的可获取性,要求待监测模块在设计时必须符合一定的规范和具有标准接口.

(1) 分组处理编码规范

PktScope 技术的实现是基于状态机的状态为展示基础的,因此,待监测模块的编码必须使用状态机编码.状态机是数字系统设计中的重要组成部分,是实现高效率高可靠性逻辑控制的重要途径[5].状态机可归纳为 4 个要素,即状态、输入、输出、条件.状态只可能在时钟跳变沿满足条件的情况下从一个状态转向另一个状态,转向下一状态还是留在原状态取决于输入,也取决于当前所在状态.分组处理过程中,状态的跳转过程反映了分组处理的过程变化,状态信息应具有可读性和可收集性.因此,分组处理编码要求每个功能模块只能实现一个处理状态机,并采用一段式状态机描述方法编写.分组处理状态机示例如图 2 所示.相对于二段式、三段式状态机编码,一段式的状态机描述方法将整个状态机写在一个 always 模块里,清晰明确,不会造成状态收集冗余混乱,更易于直观分析.

```
reg [4:0] state;
parameter state1 =...,state2 =...,
          ..... stateN =...;
always@(A or B or C)
  begin
   if(!reset)
    begin
      state  <= state 1;
    end
   else
    begin
     case(state)
      state1:
       begin
        if(...)  state <= next state;
        else     state  <= state 1;
       end
      .....
      stateN:
       begin
        if(...)  state  <= state 1;
        else     state  <= stateN;
       end
     endcase
    end
  end
```

Fig.2 Encoding example of finite state machine

图 2 有限状态机编码示例

(2) 监测接口定义

待监测模块需要向监测模块提供以下监测信息:当前状态机编码、模块的输入分组数据和输出分组数据.状态编码是监测结果最重要的展示元素,说明硬件模块内详细的分组处理过程;模块的输入分组数据是进入模块未经过状态机操作的初始分组数据;输出分组数据是完成状态机操作后的处理完毕分组数据.根据监测所需,监测接口定义见表 1,其中输入分组数据接口为常用的 FIFO 读接口,输出分组数据接口定义为常用的写 FIFO 接口.

Table 1 Monitor interface signals

表 1 监测接口信号

Name	Width (bit)	Function description
(module name)_state	5	State encoding
(module name)_rdreq	1	Enable read fifo_data
(module name)_rd_pkt	= input_pkt_width	Read input packt
(module name)_wrreq	1	Enable write fifo_data
(module name)_wr_pkt	= output_pkt_width	Write output packet

监测接口模块对应的时序关系如图 3 所示,监测部件接收输入分组数据的同时,也接收来自待测模块处理输入分组数据的状态信息.在完成一个分组的状态机处理后,根据下一个模块对该模块输出分组数据的请求信号,监测部件接收该模块的输出分组数据.

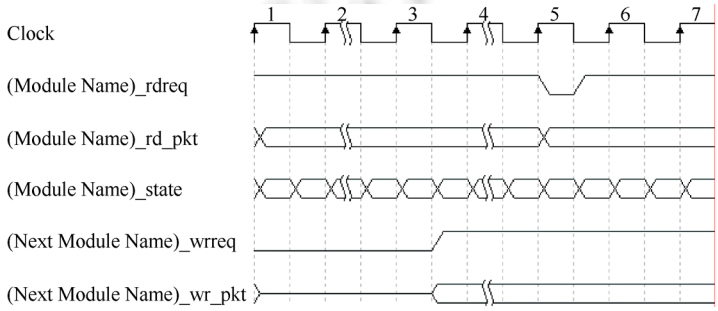

Fig.3 Timing relationship of interfact signals

图 3 接口信号时序关系

2.2.2 监测报文格式定义

PktScope 技术中,硬件负责对监测信息的收集和预处理,软件负责对信息再处理和展示.监测报文就是软硬件监测数据交互的关键数据结构.将硬件收集的监测数据处理后上传至软件进行再处理.监测报文由监测模块负责收集信息并根据软件需要的内容进行封装上传至软件.

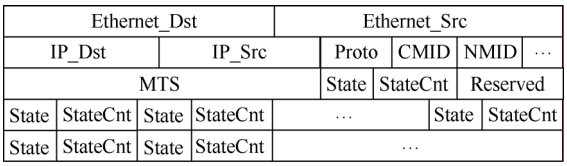

Fig.4 Format of monitor packets

图 4 监控报文格式

如图 4 所示,监测报文的内容包括:

• 当前模块编号 CMID:8 位,分组当前所在处理模块编号;

• 下一模块编号 NMID:8 位,分组将要进入的下一个模块的编号;

• 模块处理时间戳 MTS:64 位,一个状态机分组处理完成的结束时间;

• 状态编码 State:5 位,待监测模块的状态机编码(最大支持 32 个状态);

• 状态计数 StateCnt:10 位,最大支持 1 024 拍,状态机在特定状态的拍数.值得注意的是,等待分组或者其他状态可能会持续很长时间,在此情况下,监测部件实现时需保证该状态计数不溢出.

2.2.3 软件分析技术

硬件将收集到的监测信息经过部分处理后,形成监测报文上传至软件,由软件对监测报文进行解析,并形成图形化展示.软件分析的核心数据结构是监测报文链表.链表是一种常见重要的数据结构,是一种线性表[6].如图 5 所示,监测报文分别根据待监测模块号组织成多个链表.

对于链表的操作主要由两个线程负责,RX 线程和 Proc 线程.Rx 线程负责将接收到的报文挂在相应链表的尾部;Proc 线程则主要负责提取报文链表的数据按特定待监测的分组合并组织,并发送到图形展示模块.上述过程的关键在于将某一分组所经历的各个模块处理信息(多个监测报文)识别、组合,以形成对该分组的全流程监测信息集合,采用算法 1 实现.

算法 1. Software-Based packet analysis.

```
Input: thread_id, type, num
Output: pkt
while(1){
   pkt = read_pkt_from_thread(thread_id);
   if(pkt){
      switch(type){
      case IO:
         test_IO();
         break;
      case MEM:
         test_mem(num);
         break;
      case MD5:
         test_md5(num);
         break;
      }
      send_pkt(pkt,outport,pkt_len);
   }
      }
```

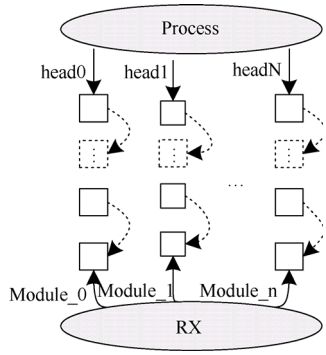

Fig.5 Structure of monitor packet lists

图 5 监测报文链表结构

3 实验分析

3.1 实验平台设置

本文基于 MagicRouter 平台对 PktScope 技术进行实验验证.MagicRouter 平台是国防科学技术大学基于 Netmagic 硬件实验平台[7,8]实现的路由器原型实验系统,主要用于网络实验的模块化可扩展软件路由器.底层的 Netmagic 硬件主要将主机/服务器平台的网络接口数目扩展到 8 个,1 口为控制端口,连接控制主机,专门用于传送控制主机对硬件的配置信息和硬件统计模块的统计信息;2 口为监测模块上传收集的分组数据及流程信息到控制主机的专用接口;3~8 口为数据出入口,进行数据传输.

MagicRouter 软件采用基于核心交换的松耦合架构(如图 6 所示),是一款架构上软硬件互补的路由器.软件部分由转发平面(接收处理进程、转发处理进程、发送处理进程)、控制平面(控制平面进程、ARP 处理进程)和平台管理(核心交换进程、配置管理进程、日志分析进程)3 部分组成.

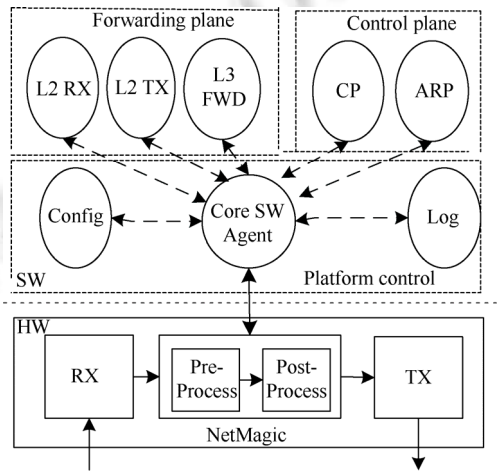

Fig.6 Structure of MagicRouter

图 6 MagicRouter 体系架构

MagicRouter 硬件部分负责接收报文并对报文添加额外头部(包含输入分组相关信息),然后交由软件处理,最后根据软件处理结果将分组输出.硬件处理功能实现在 Netmagic 平台的 UM 模块.此例以 Netmagic UM 设计指南[9]和 Netmagic 硬件开发方法[10]为依据进行设计.

Netmagic 硬件平台所实现的分组处理功能主要由两个功能模块构成:

• Parser & Lookup 模块:根据之前由软件下发的报文处理规则表以及所要监控流信息,对报文添加额外的头部后,将报文经监测模块和管理模块传送到控制主机进行查表转发;向监测模块发送数据的状态信息以及 Parse & Lookup 模块的输入和输出报文;

• Transmit 模块:接收转发信息;剥离额外头部传送给监测模块,并根据转发信息对报文进行转发;向监测模块发送数据的状态信息以及 Transmit 模块的输入和输出报文.

3.2 设计实例

我们基于 MagicRouter 平台对 PktScope 技术进行实验验证,对其硬件模块分组处理过程进行监测,逻辑架构如图 7 所示.

Parser & Lookup 模块和 Transmit 模块是分组处理的功能模块,按照第 3.2 节所述设计规范所设计,提供了相应的监测接口;监测部件包含了探测、封装、发送这 3 个模块,负责监测流,功能包括探测过滤、封装和发送.

通过对 Parser & Lookup 模块和 Transmit 模块内 FIFO 的读写请求信号的探测,接收来自待测模块的分组数据和流程信息.将这些数据封装后,组成监测报文,上传至软件.由软件进行再处理,形成图形化界面展示出来.

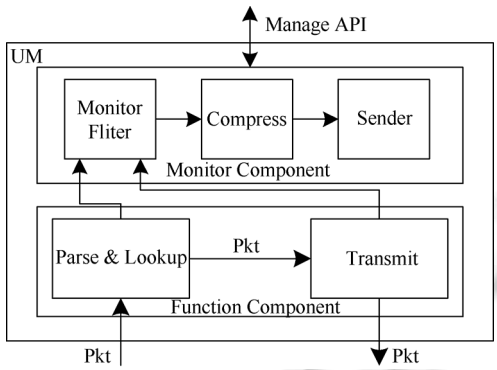

Fig.7 Abstract structure of UM module

图 7 UM 模块逻辑框架

3.3 实验结果分析

3.3.1 资源开销评估

本文在 Netmagic 平台上实现了 PktScope 技术,对 MagicRouter 的硬件模块进行监测.

Netmagic FPGA 器件采用 Arria Ⅱ GX EP2AGX45DF25C4,其含有 36100ALUTs 和 2939904Memory Bits,监测部件运行在 125MHz 上.未实现 PktScope 的 MagicRouter 系统和实现 PktScope 的 MagicRouter 系统资源占用情况见表 2.通过对比,可以发现实现 PktScope 技术对整个 MagicRouter 路由器系统资源的占有率很小,不会对整个路由器系统的性能造成影响.

Table 2 Relative resource consumption compared to without PktScope

表 2 使用 PktScope 前后相关硬件资源开销对比

Resource name	Without PtkScope	With PktScope
LC Combinationals	12 493 (35%)	15 680 (43%)
LC Registers	15 124 (42%)	17 268 (47%)
Block Memory Bits	1 146 400 (39%)	1 158 320 (39%)

3.3.2 图形化界面

如图 8 所示,该界面包含两部分:控制界面和展示界面.控制界面内是监测分组流的信息,填入后,点击监测,把监控分组流的信息下发至硬件,当流信息得到满足时,即对该分组流进行监测,监测数据经过软硬件处理后,再经界面化处理得到待测部件的结构示意图、模块内分组处理过程信息、当前模块的输入/输出分组数据.通过软件对分组数据和流程信息的处理,形成硬件分组监测处理过程说明,经图形界面化处理后,达到 PktScope 技术实现可视化的效果.

Fig.8 Graphical interface of PkeScope

图 8 PktScope 图形化界面

4 结束语

针对网络科研教学中面临的网络硬件分组处理过程监测问题,本文提出了一种基于状态机的硬件分组处理监测技术 PktScope,可实现对硬件进行分组处理过程进行实时监测,达到对网络硬件设备处理过程透明化要求,增强硬件处理过程的直观性,支撑网络设备原型系统调测试以及网络教学示范.下一步将着重研究如何实现在不改变分组信息的情况下对特定分组的监测技术.

References:

[1] Shu LL. Research and application of NS2 network simulation platform. Fujian Computer, 2015,(4):82–84 (in Chinese).

[2] Kohler E, Morris R, Chen BJ, Jannotti J, Kaashoek MF. The click modular router. ACM Trans. on Computer Systems, 2000,18(3): 263–297 (in Chinese).

[3] Altera Corporation. Using Signaltap II embedded logic analyzers in SOPC builder systems. http://www.altera.com/literature/an/an323.pdf

[4] Wan X. Application of chipscope Pro in FPGA debugging. Computer and Network, 2005,(21):58–59 (in Chinese with English abstract).

[5] Gong ST, Lü GQ, Peng LQ. The encoding mode of state machine in FPGA. Electronics Engineer, 2005,(11):56–58 (in Chinese with English abstract).

[6] Ye YJ, Tan C. Research on the application of dynamic list in C language. Network Security Technology and Application, 2014,(12): 167–169 (in Chinese with English abstract).

[7] Li T, Sun ZG, Chen YJ, Jia CB, Su Q, Guo TF. A novel packet processing model for the next-generation Internet experimental platform——EasySwitch. Chinese Journal of Computer, 2011,34(11):2187–2196 (in Chinese with English abstract).

[8] Li T, Sun Z, Jia C, Su Q, Lee M. Using net magic to observe fine-grained per-flow latency measurement. ACM SIGCOMM Computer Communication Review. 2011,41(4):466–467.

[9] Mao XL, Li T, Sun ZG. Innovation Platform for Next Generation Internet Architecture. Changsha: National University of Defense Technology Press, 2012. 92–132.

[10] Cao CZ, Mao JB, Sun ZG, Yin JB, Lin Q, Gong XL. Method of NetMagic hardware development. Computer Engineering and Science, 2014,(9):1678–1683 (in Chinese with English abstract).

附中文参考文献:

[1] 舒磊磊.NS2 网络仿真平台的研究与应用.福建电脑,2015,(4):82–84.

[4] 万翔.Chipscope Pro 在 FPGA 调试中的应用.计算机与网络,2005,(21):58–59.

[5] 龚书涛,吕国强,彭良清.在 FPGA 中状态机的编码方式.电子工程师,2005,(11):56–58.

[6] 叶勇健,谭超.C 语言中动态链表的应用研究.网络安全技术与应用,2014,(12):167–169.

[10] 曹成周,毛健彪,孙志刚,尹佳斌,林琦,龚小林.NetMagic 平台硬件开发方法.计算机工程与科学,2014,36(9):1678–1683.

厉俊男(1992－),男,浙江金华人,博士,主要研究领域为高性能路由器,网络处理器.

唐路(1988－),男,博士,主要研究领域为高性能路由器,网络处理器.

胡锴(1988－),男,硕士,主要研究领域为新型互联网体系结构,高性能路由与交换技术.

李韬(1983－),男,博士,主要研究领域为计算机网络,网络处理器.