一种基于蠕虫算法的并行DSP网络结构探测

邹修国

（南京农业大学工学院 南京 210031）

摘 要 并行DSP网络构成方法很多，将EMIF和McBSP两种方法结合起来，综合了这两种方法的优点，对于其结构可以采用蠕虫算法。网状结构里深度优先遍历的算法是一种很好的蠕虫算法，在阐述该蠕虫算法的具体实现过程后，分析得出该算法可以探测出并行DSP网络的结构。如果采用相同的DSP芯片，该算法在并行DSP网络系统中具有一定的通用性。

关键词 蠕虫算法，并行DSP网络，结构探测

中图法分类号 TP301.6 **文献标识码** A

Framework Detecting in Parallel DSP Network Based on Wormhole Algorithm

ZOU Xiu-guo

(College of Engineering, Nanjing Agricultural University, Nanjing 210031, China)

Abstract There are many ways that compose parallel DSP network. Two methods connecting DSP are EMIF and McBSP, combining the advantages of both approaches. Its structure can use wormhole algorithm. Depth-first traversal algorithm is a good wormhole algorithm. Elaborating on the concrete realization of the wormhole algorithm, and geting the framework with this algorithm in the detecting. If using the same DSP chip, the parallel DSP algorithm has more applicability in parallel DSP system.

Keywords Wormhole algorithm, Parallel DSP network, Framework detect

在视频和图像处理中，数字视频图像信号对信号处理能力的要求越来越高。在目前的技术条件下，单片DSP恐难以实现，这需要一种功能强大的实时分布式处理系统，因此有必要采用实时性强、精度高且具备高数据吞吐量的多个DSP并行处理大规模的信号数据[1]。

蠕虫算法思想就是像一个“蠕虫”一样在不断地寻找可以通过的路径，并将程序不停地复制到所找到的节点所在的DSP上，最终将程序布满所有可以到达的DSP[2]。本文以TI公司的TMS320DM6437芯片构成的并行处理网络为例，讨论一种蠕虫算法的实现。蠕虫算法的功能主要是实现探测并行DSP系统的硬件节点互联结构图，同时也找出并行DSP系统中每个节点的加载路径，以为并行应用程序的加载提供通路。

1 并行DSP网络设计

本设计采用的DM6437达芬奇处理器为TI公司新一代高性能的数字媒体处理器，是一颗工作频率达600MHz的处理器，拥有采用高性能的C64x+内核、高达600MHz的主频、短1.67ns的指令周期、每个时钟周期可并行执行8个指令等出色性能。采用TI公司DM6437并联DSP一般有3种连接接口可用：对外的主机接口HPI、外部存储器接口EMIF、多通道缓冲串口McBSP。利用DSP 3种不同接口的互连各有优缺点，HPI有利于外部主处理器对各个DSP进行控制，适合于主处理器和多个DSP构成主从方式的互连系统，如用ARM控制多个DSP处理器；EMIF数据传输的速率高，适合于构成DSP高速全互连阵列；McBSP接口简单，适用于对传输速率要求不高的低速全互连系统[3]。

1.1 利用McBSP组成多DSP互连系统

McBSP称为多通道缓冲串口，是一个双向收发端口，它有一个发送端口和一个接收端口，多个DSP可以通过McBSP连接到一个串行时际交换芯片，采用时际交换的方式进行数据交换。数据收发以帧为单位进行，每个发送帧分成n个发送时隙，不同的发送时隙对应不同的接收DSP，例如：DSP0的发送端口在时隙1给DSP1发送数据，在时隙2给DSP2发送数据，在时隙n给DSPn发送数据；每个接收帧分成n个接收时隙，不同的接收时隙对应不同的发送DSP。例如：DSP1的接收端口在时隙0接收来自DSP0的数据，在时隙2接收来自DSP2的数据，在时隙n接收来自DSPn的数据[4]。

利用DM6437 DSP的两个McBSP口可以互连成一个高速的通信通道，同型号DSP的McBSP口连接可达到50Mbps的通信速度。这种方法的优点是接口简单，可以实现多DSP的全互联。缺点是数据以串行方式通信，速率低，这也是限制整个系统速度的瓶颈[4]。

1.2 利用EMIF组成多DSP互连系统

具体做法有两种：一是两个DSP共享存储器，异步SRAM作为全局存储器由所有DSP共享，DSP访问SRAM

本文受南京农业大学青年科技创新基金(KJ2010031)资助。

邹修国(1979—)，男，博士，讲师，主要研究方向为视觉技术与模式识别，E-mail：zouxiuguo@njau.edu.cn。

的总线是 EMIF。DSP 片间通信是通过向共享 SRAM 中写入和读取数据两个过程完成。

另一种是通过 FIFO 直接互连。在多个 DSP 组成的全互连方案中，两两 DSP 之间专用的 BIFIF0(双向先进先出存储器)通过 EMIF 接口互连，DSP 各自通过 BIIFO 与主机或外设互连。这种方法的优点是能够实现 DSP 间的数据高速传输；缺点是 DSP 需查询两个 FIF0 的状态，每两个 DSP 固定的连接有时是不需要的，使用 FIFO 浪费资源，系统扩展也比较困难[3]。

本设计综合考虑了 McBSP 接口简单和 EMIF 数据传输速率高的双重优点，通过 EMIF 口与双端口 RAM 的一个端口相连，而双端口 RAM 的另一端口与 FPGA 相连，在 FPGA 内部实现两片 DSP 之间数据的交换。DM6437 拥有两个外部存储器接口(EMIFA/EMIFB)，数据总线宽度分别为 64 位和 16 位。将 EMIFA 与双端口 RAM 相连实现多 DSP 的互连，而 EMIFB 与 FLASH 相连以实现程序的 FLASH 加载[4]。具体两个 DM6437 芯片相连的硬件框图如图 1 所示。

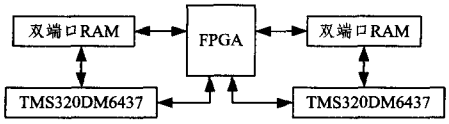

图 1　两个 DM6437 芯片相连的硬件框图

2　算法设计

并行 DSP 网络是多个 DM6437 按照图 1 的方式两两相连，由与主控机如 PC 机或者 ARM 直接相连的 DM6437 首先接受命令，然后按照算法程序依次寻找下一个 DM6437，一直到算法程序运行结束，并将结果返回给主控机。算法程序运行过程就像一个“蠕虫”一样在不断地寻找可以通过的路径，所以该算法也称之为一种蠕虫算法。

本设计所采用的蠕虫算法是用网状结构里深度优先遍历的算法，因为是 DSP 芯片，对于某一个芯片节点我们简称为 D_i，以 D_i 开始的连通图：(1)访问 D_i；(2)分别深度优先遍历 D_i 的各个未被访问的邻接点。具体过程如下：在对 D_i 做过访问标记后，选择一条从 D_i 出发的未检测过的边(D_i, D_j)。若发现顶点 D_j 已访问过，则重新选择另一条从 D_i 出发的未检测过的边，否则沿边(D_i, D_j)到达未曾访问过的 D_j，对 D_j 访问并将其标记为已访问过；然后从 D_j 开始搜索，直到搜索完从 D_j 出发的所有路径，即访问完所有从 D_j 出发可达的顶点之后，才回溯到顶点 D_i，并且再选择一条从 D_i 出发的未检测过的边。上述过程直至从 D_i 出发的所有边都已检测过为止。此时，若 D_i 不是源点，则回溯到在 D_i 之前被访问过的顶点；否则图中所有和源点有路径相通的顶点(即从源点可达的所有顶点)都已被访问过，若探测的并行 DSP 网络是连通图，则遍历过程结束，否则继续选择一个尚未被访问的顶点作为新源点，进行新的搜索过程[5]。

在算法运行过程中，必须知道顶点是否已经被访问过，我们用一个全局数组 visited[]来记录顶点是否被访问过。如果 visited[i]的值为 1，则顶点 D_i 已经被访问，否则没有被访问。

主要算法程序如下。

```
int visited[MAXSize];
typedef struct listnode
{
  int adjvex;
  struct listnode * next;
  }Listnode;
typedef struct
{
  int data;
  listnode * first;
  }Headnode;
typedef struct
{
  Headnode vexs[MAXSize];
  int vexnum;
  int arcnum;
  }Graph;
Void DFS(Graph G,int D)
{
  visited[v]=1;
  cout<<v;
  For(D 的每一个邻接点 X)
  {If(visited[X]==0)//如果没有被访问过
  DFS(G,X)}
}
```

3　算法分析及结果

算法在探测网络结构过程中主要就是由主控机(如 PC 机或者 ARM)发出探测命令，由与主控机直接相连的 DM6437 首先接受命令，运行已经存储在片内存储器上的算法程序，按照算法程序寻找下一个 DM6437，一直到算法程序运行结束，并将结果返回给主控机。经过程序运行，可以得到树状层次结构，如图 2 所示。在该图中，双圈圆是探测到的 DSP 芯片尚未在 visited 数组中被记录访问过的芯片，单圈圆表示探测到的 DSP 已经被访问过一次了。

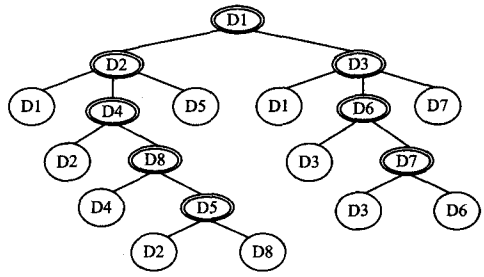

图 2　采用 DFS 蠕虫算法得到的树状结构

由 DFS 蠕虫算法得到的树状结构可以得到生成的一种遍历顺序为：D1→D2→D4→D8→D5→D3→D6→D7，以这一顺序可以进一步得到并行 DSP 网络的结构。具体做法：DFS 蠕虫算法生成树中，除了根节点外，每一个节点孩子中必有一个是父亲节点，称为父亲孩子节点，此节点必然是单圈圆表示，应该连同生成它的分支删去；某节点的孩子节点在其它孩子节点的子孙中以双圈圆出现，该孩子节点必然是单圈圆表示，应该与以双圈圆重复出现的其它孩子的子孙节点相合并，同时删除重复的父亲节点；最后得到的图形结构应该只包含双圈圆。针对图 2，按照以上具体做法列一张表，具体表示如表 1 所列，其中 0 表示没有符合条件的节点。

(下转第 88 页)

M2，此次握手失败。

在这样设计的情况下，在入侵者不知道 B 的私钥或者 A 的公钥的情况下，握手不会成功，保证了下一步密文传输的安全。

2.2 密文传输子协议的设计

密文传输子协议用于 NIDS 的加密数据数据传输，并在传输过程中保证数据的机密性和完整性。密文传输协议具体设计如下所示：

M3：$A->B$：$\{A,Nb\}Kb,\{M'\}Kab,M''$

在本密文传输协议中，M' 表示传输的 NIDS 数据，M'' 表示用于加密 NIDS 数据的数字签名。在握手协议中 A 成功地验证完消息 M2 后，首先要对将要进行传输的 NIDS 数据 M' 利用会话密钥 Kab 进行加密，然后利用自己的私钥对加密后的数据进行数字签名生成 M''；最后利用 B 的公钥将自己的标识 A 和 B 产生的随机数 Nb 一起加密，并连同加密后的 M' 和 M'' 信息一起传输给 B。B 收到 A 的消息 M3 后，首先验证 Nb 是否正确，然后验证 NIDS 数据的完整性和 NIDS 数据的机密性。

在对 NIDS 数据进行加密和数字签名时，采用的是 XML 加密和 XML 签名技术。XML 加密技术与传统的加密技术最大的区别就是，传统加密技术是对整个数据进行加密，而 XML 加密技术不仅继承了传统机密技术的优点，更可以对单个元素进行加密，这样就灵活性方面要远远高于传统加密技术。因此利用 XML 加密技术的灵活性的特点，仅仅对 NIDS 数据中的较敏感数据进行部分加密，而对其它非敏感数据采用 XML 签名技术，使得数据的机密性既得到了保证，又可以最大限度地减轻系统加密解密的开销[2]。

3 协议的验证

协议的安全性的验证有两种方法：一种是采用模拟攻击的检测方法，通过对各个子协议进行攻击来检验其安全性；另外一种是采用形式化分析方法，运用形式化语言对协议进行安全性分析。其中形式化方法已经被证明是一种强有力的系统分析和验证技术，已经得到了广泛的应用。

本文采用的是 SPIN 系统验证工具对安全协议进行协议的安全性验证。利用建模语言 Promela，对安全协议进行形式化建模后，SPIN 作为模型检测器具体对协议的具体验证步骤是[3]：

(1) 写出需要验证的系统属性要求，用 LTL(Lin2ear Temporal Logic)方程描述；

(2) 利用 SPIN 对系统属性进行验证；

(3) 若属性为假，SPIN 会生成一个 trail 文件，利用该文件进行引导仿真，跟踪协议运行过程，找出攻击序列；

(4) 否则系统属性为真，验证结束。

可以看出，形式化建模是协议分析的非常关键的一步。对协议描述中的 A、B 以及入侵者进行形式化建模，模型如图 2 所示。

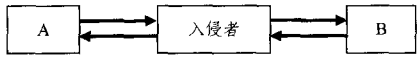

图 2 安全协议的 SPN 模型

安全协议的目的是在加密状态下确保协议主体的相互鉴别，换而言之，如果 A 和 B 成功地运行了一次协议，那么 A 的响应对象是 B，而且 B 的请求对象是 A。会话密钥不会被第三方窃取，也就是说如果 A 成功地与 B 完成了一次协议的运行，则入侵者不可能知道会话密钥 Kab。

结束语 通过对各种网络入侵检测系统通信安全问题的研究，提出了一种新的网络入侵检测系统的安全通信协议，并在理论上进行了验证。如何使得安全性和通信效率并重，是网络入侵检测系统通信协议设计的关键。该协议不使用第三方网络安全协议，不仅使得通信协议的效率得到了提高，又消除了第三方安全协议带来的安全隐患，使得安全性和通信效率得以平衡。

参 考 文 献

[1] 魏兵役. 网络入侵检测系统的分析与研究[J]. 信息与电脑，2010,4

[2] 吴启明. XML 安全加密技术框架[J]. 电脑知识与技术，2007,24

[3] 陈性元，杨艳，任志宇. 网络安全通信协议[M]. 北京：北京高等教育出版社，2009

(上接第 77 页)

表 1 各节点互联信息表

探测的 DSP	父亲孩子节点	孩子节点与其它孩子子孙节点重复
D1	0	0
D2	D1	D5
D4	D2	0
D8	D4	0
D5	D8	0
D3	D1	D7
D6	D3	0
D7	D6	0

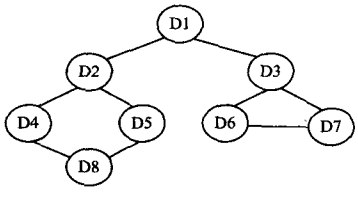

图 3 探测得到的并行 DSP 网络结构

根据上面针对图 2 的改造步骤及表 1 的详细信息，最后得到的改造图如图 3 所示，即是要探测的并行 DSP 网络结构。

结束语 本文针对并行 DSP 系统，设计一种蠕虫算法探测其结构，并分析算法的正确性。本文在蠕虫算法中所提到的基于深度优先历递归法寻找节点和构造结构的具体步骤都是可以实际运用的。随着电子技术的迅猛发展，并行 DSP 技术将大规模地应用到数字信号处理的各个领域。

参 考 文 献

[1] 王哲，王希敏. 并行 DSP 系统消息传递路由算法[J]. 计算机工程，2009(17)：241-243，246

[2] 徐精华，邹雄，王旭成. 基于蠕虫算法的 DSP 网络结构探测[J]. 计算机与现代化，2010(1)：16-18，22

[3] 任骊平，陈王骞. 多 DSP 系统互连方案分析[J]. 电子技术应用，2002(04)：50-52

[4] 林晓静. 基于 TMS320C6416 的并行 DSP 板的设计与实现[D]. 南京：南京理工大学，2007

[5] 严蔚敏. 数据结构(C 语言版)[M]. 北京：清华大学出版社，1997