南开大学

Nankai University

南　　开　　大　　学

计　算　机　学　院

并行程序设计调研报告

对神威·太湖之光及其并行体系结构的研究

姓名：徐海濛

年级：2022 级

专业：计算机科学与技术

指导教师：王刚

2024 年 3 月 17 日

摘要

本文首先对国内和国际的超算发展历程进行了调研，分析了从 1929 年到如今超级计算机的演变，对当今中国超算实力和超算的世界格局进行了调研，并由此得出超算的发展对于国家有重大意义。然后对神威·太湖之光超级计算机进行了各个角度的调研，重点分析了其采用的国产自研的申威体系架构。在调研的基础上，引出了对于并行计算模型的研究和了解，学习并调研了几个常用的并行计算模型：PRAM，BSP，LogP。另一方面，异构架构下的程序编写需要使用并行的编程框架。本文对其中一种常用的兼容性较高，能够为不同并行体系提供接口的框架，OpenCL 进行了调研与学习。

关键字：神威·太湖之光 并行计算 OpenCL

目录

(一) 超级计算机的发展：中国与世界

1.1 国际超算发展历程

从 1929 年《纽约世界报》首次提出“超级计算 (Supercomputing)”的概念开始，到如今拥有千万亿次浮点计算能力的超级计算机的诞生，超级计算机的发展历程经历了许多里程碑事件。

二战期间，超级计算机首次被用于军事用途，英军在布莱切利园成立了专门破解密码的机构，并引入了世界上第一台电子计算机 Colossus 来解读德军的电传密码。1961 年，IBM 推出了超级计算机 STRETCH，1964 年，美国科学家西蒙·克雷研制成功了 CDC6600，开启了超级计算机的新时代。CDC6600 成为了当时最快的计算机，其性能远超同期的商用机。这台机器具有 8MB 内存、运算速度达 300 万次，是当时其他计算机的 10 倍。CDC6600 的出现开启了超级计算机的新时代，为后来的超级计算机技术发展奠定了基础。

随着超级计算机技术的不断发展,向量计算机、对称多处理 (SMP) 和大规模并行处理 (MPP)、集群系统等不同类型的超级计算机相继出现。1974 年，CDC 推出了第一台向量机 STAR-100，标志着向量处理器的问世。1976 年，克雷公司推出了 Cray-1，采用了流水线结构和向量处理技术，成为了当时性能最强大的超级计算机。80 年代至 90 年代，超级计算机的发展经历了大规模并行运算系统的兴起，以及各种新技术的应用，如 Linux 系统、VLIW 体系结构等。1993 年，德国曼海姆大学创建了全球超级计算机 TOP500 排名榜，成为了评价超级计算机性能的国际标准。2003 年开始，Linux 系统逐渐成为超级计算机的主流操作系统。

2000 年以后，超级计算机的性能不断提升，中国和美国成为超级计算机领域的主要竞争对手。2016 年，中国“神威太湖之光”问世，成为了世界上第一台超过十亿亿次 (100PFlops) 量级峰值计算能力的超级计算机，引领了超级计算机技术的新发展方向。

超级计算机的发展不仅推动了科学技术的进步，也在军事、天气预报、基因研究等领域发挥着重要作用。随着技术的不断创新和应用，超级计算机将继续在人类社会的各个领域发挥着重要作用，为人类的未来发展提供强大支持。

1.2 中国超算发展历程

自上世纪 50 年代初，中国正式进入电子计算机领域起，中国超级计算机事业经历了一段令人瞩目的发展历程。1956 年，中国成功仿制出苏联 M-3 大型计算机，实现了从零到一的重要突破，奠定了中国在电子计算机领域的基础。随后，中国陆续推出了多款自主设计的计算机，如 1960 年的 107 机和 119 机，以及 1965 年的 109 乙机，这些计算机在当时的国防工程和科学计算中发挥了重要作用。然而，整个六十年代，由于国内计算机研发都是围绕重大国防工程进行，只追求不断提高运算速度，对计算机整体性能和普及性考虑并不多，不仅资金花费巨大，也忽视了社会生产建设需求，更没有批量生产的概念。

1973 年 8 月，我国首台百万次集成电路大型计算机 150 机诞生，这台计算机主内存 130K。配有多个程序和操作系统，每秒运算速度达 100 万次。从这时起，中国大型计算机逐渐转移到经济建设层面，在中国石油勘探、气象预报、科学计算等领域肩负起新使命。1978 年，邓小平明确提出：中国必须拥有自己的超级计算机，并启动了“785 超级计算机”工程。1983 年，中国成功研制出了第一台亿次超级计算机“银河一号”，标志着中国成为继美国、日本之后，第三个拥有自主设计和制造超级计算机的国家。此后，中国超级计算机事业蓬勃发展，不断推出更加强大的超算产品，如 1993 年诞生的“曙光一号”，以及 2008 年的曙光 5000 型超级计算机，其峰值运算速度达到每秒百万亿次，刷新了中国超级计算机的纪录。

随着时间的推移，中国超级计算机不仅在国防和科学计算领域取得了重大突破，而且开始逐步向民用领域渗透，如辽河油田的应用就是一个典型例子。此外，在中国超级计算机的不断演进过程中，国产化程度也不断提高，国产化的超级计算机已成为中国科技创新的重要支撑力量。

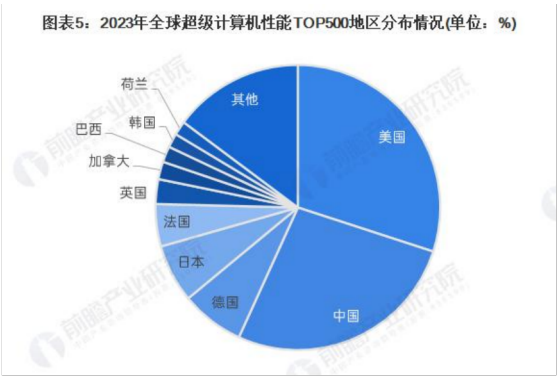

图 1

2018 年，曙光、天河与神威已进入到超级计算机竞赛领域的 E 级（秒钟运算一百亿亿次）超算研发，并逐步实现 CPU 和加速器的全国产化。2021 年，第五十八届全球超级计算机 TOP500 排行榜中，中国超级计算机有 173 台进入榜单，占比 34.6 %。第二名的美国为 149 台，占比 29.8 %。2022 年上半年的全球超算 TOP500 榜单中，中国的神威太湖之光排名第六，已是“三剑客”中成绩最好的一家。然而，尽管中国超级计算机取得了巨大成就，但与全球顶尖超级计算机的差距仍然存在，这也是中国超级计算机未来发展的重要挑战之一。

（二） 神威·太湖之光并行体系架构调研

神威·太湖之光超级计算机是由国家并行计算机工程技术研究中心研制的世界上首台运算速度超过十亿亿次的超级计算机。如何在高性能计算应用中发挥超级计算机潜在的计算能力一直是高性能研究的主要挑战之一。并行计算是提高计算性能的重要方法，通过研究神威·太湖之光的并行体系架构，可以加深对于超级计算机原理的理解和当今最新的并行体系架构的了解。

2.1 申威体系架构

作为国产自主研发的芯片“申威 26010”，是国产芯片的重大突破，该芯片是异构众核处理器 [9-10]（如图 1）。目前，主流的高性能计算机大多采用了“主处理器 + 加速部件”的异构架构, 如 Summit、Sierra[1] 超级计算机, 均使用 NVIDIA 公司的 Tesla V100[2] 作为加速部件, 旨在实现更高的性能功耗比和计算密度。申威众核处理器采用片上融合的异构体系结构，由 4 个异构群构成，每个异构群包括一个主核、64 个从核构成的从核簇、异构群接口和存储控制器，整芯片共 260 个计算核心，众核处理器还集成系统接口总线，用于连接标准 PCIe 接口实现片间直连和互连，管理与维护接口实现系统管理、维护与测试。四个异构群和系统接口总线通过群间传输网络实现存储共享和通信。

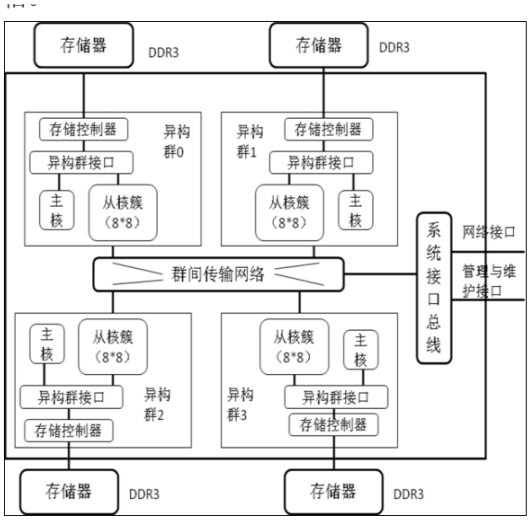

图 2: 申威众核处理器结构图

根据洪文杰，李肯立（2017）等人的研究，将主从核的异构体系模型分为了以下四种：

1. 主从加速并行模式：主核主要完成无法用众核并行部分的计算以及通信，而在从核进行任务计算时，主核等待。
2. 主从协同并行模式：主核和从核作为对等的个体进行并行计算，根据各自计算能力进行负载分配，共同完成核心段的计算。
3. 主从异步并行模式：在从核进行加速计算的同时，主核完成其他计算、通信或 I/O 等操作，提高主从协作的并行效率。
4. 主从动态并行模式：主核负责任务分配，从核负责取得新计算任务、完成计算、写回计算结果。

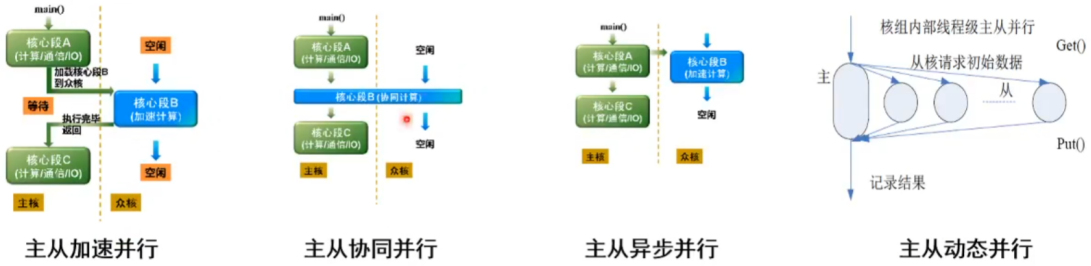

图 3: 四种主从并行模式流程图

申威架构在核组间利用 MPI（Message Passing Interface）实现进程级并行，在核组内采用 Athread 并行编程模型实现线程级并行，利用运算核心阵列实现主从加速并行方式。程序的核心段通过 Athread 模型开启线程组，并加载到运算核心阵列上进行加速计算。在运算核心阵列进行核心段运算过程中，通用管理核心处于等待状态，直至运算核心阵列完成该核心段的计算任务。由于申威平台在线程组开启时开销较大，因此我们将尽可能保证并行粒度最大化，即尽可能实现外层循环的并行执行。

申威架构的核心为国产 SW26010 异构众核处理器。SW26010 是面向高性能计算领域开发的处理器，采用片上计算阵列集群和分布式共享存储结构相结合的异构架构，使用定制的申威指

令集，是” 神威·太湖之光” 得以实现高计算速度和低功耗的核心部件。SW26010 处理器芯片主要由 4 个核组、片上互联网络和系统接口组成。核组是系统进行计算资源管理的基础单位，包括一个通用管理核心、一个运算核心阵列和存储控制器等。每个运算核心阵列包含 64 个精简运算核心，采用 8×8 的阵列通信网络进行连接。存储结构方面，SW26010 处理器单个芯片上 4 个核组的物理空间统一编址，通用管理核心和运算核心均可以访问芯片上的所有主存空间。单个核组的存储系统结构包括通用管理核心拥有 L1 和 L2 两级硬件 Cache，运算核心存储为采用 SPM 结构的 LDM，LDM 空间大小为 64KB。运算核心可以通过 Athread 模型加载 DMA 命令实现 LDM 和主存之间的批量数据传输。DMA 传输的效率与传输的数据量、DMA 命令数量、数据在内存中的连续性以及 DMA 传输方式等密切相关，硬件同时支持阻塞 DMA 传输与非阻塞 DMA 传输。虽然通用管理核心与运算核心都可以访问主存和 LDM，但处理器核心访问不同存储器的延迟有很大差别，运算核心访问 LDM 的延迟远低于访问主存的延迟，而芯片面积及功耗的限制导致 LDM 的空间有限。因此在实际应用中，运用运算核心进行加速计算时，如何充分利用访问时延短的 LDM 是发挥 SW26010 处理器性能优势的关键。

除此之外，申威芯片采用了国产自研的指令集。自主指令集是国产处理器冲破国外同行业的技术封锁和知识产权壁垒的基础，申威２６０１０处理器的２类核心采用申威自主６４ｂ的ＲＩＳＣ指令集，运算控制核心和运算核心的基础指令集保持兼容，支持８ｂ，１６ｂ，３２ｂ和６４ｂ整数运算、单精度和双精度浮点运算，并根据高性能应用需求进行了扩展：２类核心均支持２５６ｂ的ＳＩＭＤ扩展指令，支持整数和浮点的短向量操作，使得运算控制核心每个时钟周期最快可以完成１６个双精度浮点运算，运算核心每个时钟周期最快可以完成８个双精度浮点运算

2.2 并行计算模型

并行计算模型作为超级计算机的系统设计者和应用开发者之间的重要桥梁，从大规模应用并行算法设计需求出发，将真实并行计算机系统的计算、访存、通信等基本特征参数化抽象成计算模型，为并行计算提供硬件设计和应用开发的接口。因此，我们调研了常应用的几种并行计算模型和相关的研究，以加深对

2.2.1 PRAM 模型

当谈到并行计算时，我们首先要了解 PRAM 模型。这个模型描述了一种处理器内部的并行计算方式，其核心思想是多个处理核心在共享存储结构下同时执行计算任务。这种模型的出现是为了解决单核处理器在计算性能上的瓶颈，通过同时利用多个处理核心来提高计算速度。PRAM 模型将不同的计算核心视为一个整体，它们共享同一块内存，可以同时读取和写入数据，从而实现高效的并行计算。PRAM 模型的优点是特别适合于并行算法的表达、分析和比较，使用简单，很多关于并行计算机的底层细节，比如处理器间通信、存储系统管理和进程同步都被隐含在模型中；易于设计算法和稍加修改便可以运行在不同的并行计算机系统上；根据需要，可以在 PRAM 模型中加入一些诸如同步和通信等需要考虑的内容。缺点是 PRAM 模型是同步的，这就意味着所有的指令都按照锁步的方式操作，用户虽然感觉不到同步的存在，但同步的存在的确很耗费时间，而且不能反映现实中很多系统的异步性。同时，未能描述锁线程技术和流水线预取技术，而这两种技术又是当今并行体系结构用的最普遍的技术。

2.2.2 BSP 模型

接着，我们需要了解 BSP 模型，主要应用于分布式存储结构的多机并行系统。BSP（Bulk Synchronous Parallelism）模型又称为块同步并行模型，该模型是由哈佛大学的 Leslie Valiant 提出的一种基于超级步和全局“栅障”同步的并行模型，对高性能领域的发展起到了不可估量的作用。在 BSP 模型中，整个计算过程是由一系列使用全局同步分开的周期为 L 的计算部分组成，

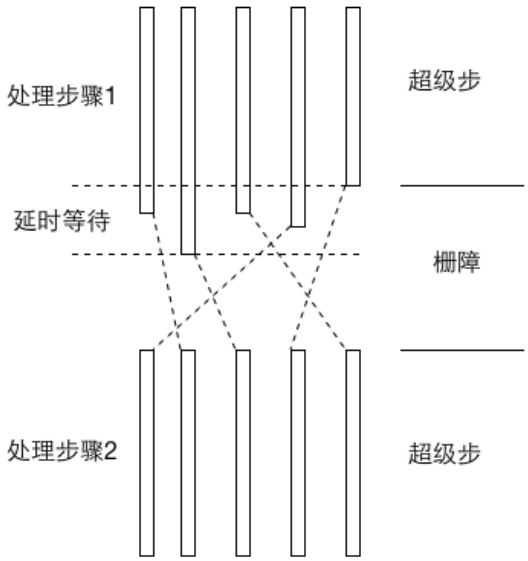

图 4: Enter Caption

这些计算部分称为超级步（Super Step）。在各个超级步中，整个系统中的每个处理器负责完成局部的计算任务，并利用选路器进行接受和发送消息，使消息传输到正确的工作节点以作为下一步的输入数据，此后系统进行一次全局检查，以保证该超级步的执行已在所有的处理器上全部结束。在 BSP 的一个超级步中，每个进程的执行过程均包括以下三种操作：本地计算操作，进程通信操作和栅障同步操作。BSP 模型中多进程并行工作的方式使其能够对流式计算中源源不断流入的数据进行持续和多级的处理，但由于 BSP 模型在设计中存在栅障的概念（如图 1-1 所示），此时若数据处理所花费的时间不同，则在整个系统中执行较快的进程会由于较慢的进程未执行结束而进行不必要的等待，产生不必要的等待延时 T。对于要求低延时的流式计算系统来说，该等待延时将影响流式计算系统的性能，因此 BSP 模型并不能满足流式计算的应用需求。

2.2.3 LogP 模型

LogP 模型是由 Culler(1993) 提出的，是一种分布存储的、点到点通信的多处理器模型。其中通信由一组参数描述，实行隐式同步。LogP 模型的通信网络由 4 个主要参数来描述

L(Latency): 表示源处理器与目的处理器进行消息 (一个或几个字) 通信所需的等待或延退时间的上限，表示网络中消息的延迟。o(overhead): 表示处理器准备发送或接收每条消息的时间开销 (包括操作系统核心开销和网络软件开销)，在这段时间内处理器不能执行其他操作。g(gap): 表示一台处理器连续两次发送或接收消息时的最小时间间隔，其倒数即微处理器的通信带宽。P(Processor): 处理器/存储器模块个数

在 LogP 模型中，更关注的是处理器间的网络性能和处理器本身的计算性能。通过这些参数，我们可以预估算法的执行时间，以便更好地优化并行计算任务的执行顺序和方式。模型的特点是抓住了网络与处理器之间的性能瓶颈。g 反映了通信带宽，单位时间内最多有 Lg 个消息能进行处理器间传送。每台物理处理器可以模拟多台虚拟处理器 (VP)，当某台 VP 有访问请求时，计算不会终止，但 VP 的数目受限于通信带宽和上下文交换的开销。VP 受限于网络容量，最多有 Lg 台 VP。LogP 同样存在一些不足，模型主要适用于消息传递算法设计，对于共享存储模式，则简单地认为异地读操作相当于两次消息传递，未考虑流水线预取技术、Cache 引起的数据不一致性及 Cache 命中率对计算的影响。

2.3 异构并行编程框架 OpenCL

异构架构下的程序如何编写，是异构系统需要解决的重要问题。ＮＶＩＤＩＡ提出了ＣＵＤＡ，微软提出了Ｃ＋＋ＡＭＰ，ＩＢＭ提出了ＬＩＭＥ，Ｉｎｔｅｌ提出了Ｍｅｒｇｅ等等．然而，这些并行编程模型都只适用于特定的异构系统，出现了不同异构系统需要使用不同编程模型的问题。ｈｒｏｎｏｓ　Ｇｒｏｕｐ首次提出了通用的异构并行编程框架ＯｐｅｎＣＬ，为各种不同的异构系统提供统一的并行编程接口，获得众多主流处理器厂商的支持。伍明川, 黄磊 (2018) 等人研究了在申威体系架构上实现对ＯｐｅｎＣＬ编程框架的支持。

在ＯｐｅｎＣＬ编程框架中，我们首先需要一个主机处理器（Host），一般是 CPU。而其他的硬件处理器（多核 CPU/GPU/DSP 等）被抽象成 OpenCL 设备（Device）。每个设备包含多个计算单元（Compute Unit），每个计算单元又包含多个处理单元（Processing Element）。Device 对应我们上面提到的 Adreno GPU 和 Mali GPU，计算单元 CU 对应 Adreno GPU 中的 SP 和 Mali GPU 中的 Shader Core，而处理单元 PE 可以对应 SP 和 Shader Core 中的运算单元。

在执行中，主要的流程为 Host 端发送数据和任务给 Device 端，Device 端进行计算，最后在 Host 端进行同步。程序员的编程视图由平台模型、内存模型和执行模型构成．其中，ＯｐｅｎＣＬ平台模型刻画了程序员眼中的硬件视图，是实际平台的高度抽象，使不同厂商的平台能在系统中共存；ＯｐｅｎＣＬ内存模型定义了程序执行时内存的结构、内容和行为，使程序员无需考虑实际的底层内存架构；ＯｐｅｎＣＬ执行模型描述了程序的执行行为，包括并行模式、线程组织、同步操作等。

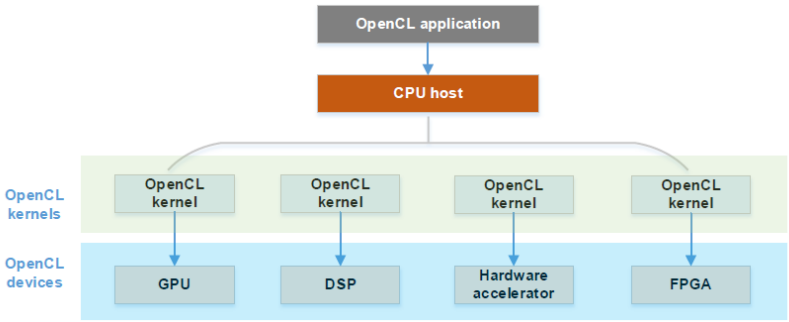

图 5: OpenCL 编程模型

平台模型中，异构系统包含一个主机（ｈｏｓｔ）和多个加速设备（ｃｏｍｐｕｔｅｄｅｖｉｃｅ）．运算部件包括ＣｏｍｐｕｔｅＵｎｉｔ（CU）和ＰｒｏｃｅｓｓＥｌｅｍｅｎｔ（ＰＥ）：每个加速设备被划分为数个对等的ＣＵ，而每个ＣＵ又被划分为数个对等的ＰＥ．程序中的计算全部执行在ＰＥ上，ＰＥ是独立参与计算的最小单元，同一个ＣＵ中的所有ＰＥ可以沿完全相同或不同的控制流路径执行．ＯｐｅｎＣＬ程序通过ｈｏｓｔ提交命令来驱动ｃｏｍｐｕｔｅｄｅｖｉｃｅ进行并行计算．

内存模型中，ＯｐｅｎＣＬ平台上的内存区域可以划分为主机端可直接访问的ｈｏｓｔｍｅｍｏｒｙ以及加速设备可直接访问的ｄｅｖｉｃｅｍｅｍｏｒｙ，ｄｅｖｉｃｅｍｅｍｏｒｙ又可划分为数个内存区域：ｇｌｏｂａｌｍｅｍｏｒｙ、ｃｏｎｓｔａｎｔｍｅｍｏｒｙ、ｌｏｃａｌｍｅｍｏｒｙ、ｐｒｉｖａｔｅｍｅｍｏｒｙ．其中ｇｌｏｂａｌｍｅｍｏｒｙ是加速设备上所有ＣＵ包含的所有ＰＥ均可访问的内存区域；ｃｏｎｓｔａｎｔｍｅｍｏｒｙ是ｇｌｏｂａｌｍｅｍｏｒｙ中的一块区域，具有只读性质；每个ＣＵ包含一块对内部所有ＰＥ都可见的ｌｏｃａｌＭｅｍｏｒｙ；每个ＰＥ具有私有的ｐｒｉｖａｔｅｍｅｍｏｒｙ．

在理解 OpenCL 的执行模型时，我们需要知道如下几个概念：Context（上下文）：每个 device 对应一个 Context，Host 端通过 Context 与 Device 端进行交互和管理；Command Queue（命令

队列)：Host 端对计算设备进行控制的通道，推送一系列的命令让 Device 端去执行，包括数据传输、执行计算任务等。一个命令队列只能管理一个设备，可以顺序执行也可以乱序执行；Kernel Objects （内核对象)：OpenCL 计算的核心部分，表现为一段 C 风格的代码。在需要设备执行计算任务时，数据会被推送到 Device 端，然后 Device 端的多个计算单元会并发地执行内核程序，完成预定的计算过程；Program Objects（程序对象)：内核对象的集合，在 OpenCL 中使用 clprogram 表示程序对象，可以使用源代码文本或者使用二进制数据来创建。

这个编译系统旨在将 OpenCL 源代码转换为适用于 SW26010 处理器的本地加速编程库，并生成可执行的目标文件。下面是整个编译过程的详细描述：

1. 词法分析和语法分析：首先，编译系统对输入的 OpenCL 源代码进行词法分析和语法分析，以生成相应的抽象语法（AST)。这一步是将源代码转换为计算机可以理解和处理的结构化表示的关键步骤。

2. OpenCL-athead 转换模块：在这个模块中，编译系统将 OpenCL 源码的 AST 转换为适用于加速编程库的 AST。这一步的主要任务是将 OpenCL 平台、内存模型和执行模型映射到 SW26010 处理器上，并通过代码变换来表达这些映射。通过这种转换，编译系统可以确保 OpenCL 代码在 SW26010 处理器上的正确执行。

3. 加速线程库代码生成模块：接下来，转换后的 AST 被进一步处理，转换为调用 athread 库的源级别代码。这些代码分解为控制核心代码和计算核心代码，以便在 SW26010 处理器上进行并行执行。这个模块的关键技术在于扩充现有 AST 的表达，以表达 athread 语义，并建立与具体 athread 函数的对应关系。

4. 本地编译器 SWCC：最后，编译系统调用本地编译器 SWCC，将控制核心代码和计算核心代码编译为可执行文件。这个步骤将确保最终生成的目标文件能够在 SW26010 处理器上正确运行，并实现 OpenCL 代码所描述的计算任务。

通过这个编译过程，编译系统能够有效地将 OpenCL 代码转换为 SW26010 处理器的本地加速编程库，并生成可执行的目标文件，为用户提供了在 SW26010 处理器上进行高效计算的平台。

参考文献

[1] 洪文杰, 李肯立, 全哲,et al. 面向神威·太湖之光的 PETSc 可扩展异构并行算法及其性能优化 [J]. 计算机学报, 2017, 40(9):13.DOI:10.11897/SP.J.1016.2017.02057.

[2] 伍明川, 黄磊, 刘颖等. 面向神威·太湖之光的国产异构众核处理器 OpenCL 编译系统 [J]. 计算机学报,2018,41(10):2236-2250.

[3] 高剑刚, 刘鑫, 李芳, 等. 面向神威众核超算系统的并行计算模型研究 [J]. 计算机学报, 2023, 46(7):1339-1349.

[4] 胡向东, 柯希明, 尹飞, 等. 高性能众核处理器申威 26010[J]. 计算机研究与发展, 2021, 58(6):11.DOI:10.7544/issn1000-1239.2021.20201041.

NKU