



Analog Devices公司DSP器件和开发工具的使用技术指南

联系中国DSP免费技术支持中心：(800) ANALOG-D 或发邮件：dsp.support@analog.com

查看我们的在线资源：<http://www.analog.com/dsp> and <http://www.analog.com/dsp/EZAnswers>

从ADSP-TS101S TigerSHARC®处理器移植代码到 ADSP-TS201S

TigerSHARC 处理器的一些考虑

Stephen Francis 供稿

2003年9月

简介

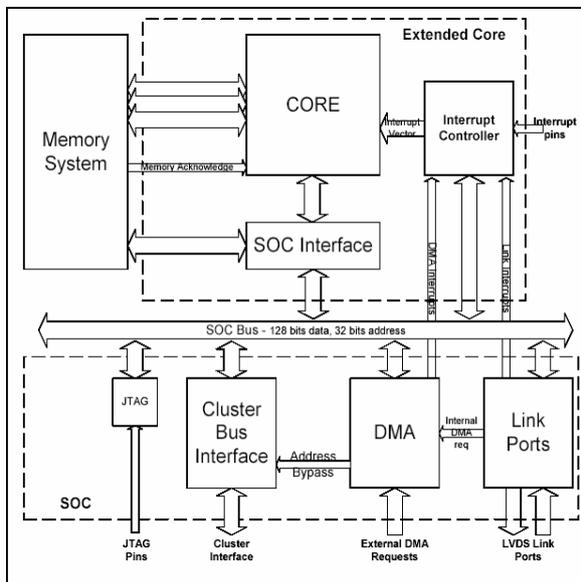


图1: ADSP-TS201S 功能块方框图

ADSP-TS20xS TigerSHARC® 处理器是 TigerSHARC 系列的第二代产品。ADSP-TS20xS TigerSHARC 处理器芯核与以前的 TigerSHARC 处理器芯核相类似，应用对象与 ADSP-TS101S 相似，如无线基站，图像处理 and 军事。

该文章讨论 ADSP-TS201S 和 ADSP-TS101S 这两种 TigerSHARC 处理器的不同之处，并将现有的为 ADSP-TS101S 而开发的代码输入到 ADSPTS201S 所必须考虑的一些东西。以下是

描述寄存器映射图变化，程序定序器，存储器 and 高速缓冲存储器，直接存储器访问 (DMA) 编程，通讯逻辑单元 (CLU)，和指令集变化。本文档也包含了强调嵌入式处理器的新型 ADSPTS20xS 系列的三部分 (ADSP-TS201S, ADSPTS202S 和 ADSP-TS203S) 的区别的一个专题讨论部分。

ADSP-TS201S 和 ADSP-TS101S 的功能性区别

ADSP-TS201S 配置高达 600MHz 的芯核时钟，总体的运算性能可达每秒可完成 4800M 次 16-bit 数乘累加。除了处理器芯核，ADSP-TS201S 包括了 24 Mb 片上动态 RAM (DRAM) 存储器，一个群集总线接口，该接口包括同步动态 RAM (SDRAM) 控制器；4 个带有低压差分信号 (LVDS) 链接的链接通道对和 14 个 DMA 通道。

图1给出了 ADSP-TS201S 的功能块方框图。处理器的功能块方框图分为两个主要部分：扩展芯核和片上系统 (SOC) 部分。扩展核心包括 DSP 芯核，中断控制器，SOC 接口，和 24 Mb 片上 DRAM。SOC 包括外部群集总线接口，链接端口，DMA 控制器和 JTAG 仿真端口。

ADSP-TS201S 是 TigerSHARC 系列的一个功能变体，与以往处理器相比，芯核内部单元，总线系统，存储器组织和外部链接都有所加强。因为 ADSP-TS201S 和 ADSP-TS101S 处理

器的芯核功能上是相同的，都可执行相同指令，ADSP-TS201还能执行一些额外的指令，如XCORRS操作。

寄存器映射图变化

扩展芯核的变化包括寄存器映射图的调整。一些寄存器在SOC代理上，其它的则在芯核内。寄存器映射图变化在下面部分重点论述，位定义寄存器变化的总结表（表1）在本节最后给出。了解寄存器映射图的更多详细信息，请参考ADSP-TS201 TigerSHARC 处理器硬件参考手册里的存储器 and 寄存器映射图章节（第二章）和与TigerSHARC VisualDSP++™ 开发工具（路径目录为 C:\ProgramFiles\Analog Devices\VisualDSP\TS\include）一同安装的系统头文件defTS201.h。注意：系统头文件，如defTS201.h，是用来声明全局定义，特别是存储器映射寄存器，系统结构和处理器。

定序寄存器文件

原来是ADSP-TS101S上的定序器一部分的一些寄存器现变为ADSPTS201S上的中断控制器的一部分了。定序器里剩下的一部分寄存器还有一些位定义的变化，一些新的为实现标志管脚操作的寄存器被加到ADSP-TS201S上的定序器上。

移到中断控制器模块的寄存器是 ILAT（低和高的，带设置和清除地址的）。IMASK（低和高的），TIMER（0和1，低和高的）和TMRIN 寄存器（0和1，低和高的）。ILAT, IMASK 和 PMASK 寄存器，除少数例外，和ADSP-TS101S上的有相同的位定义。从图2 和 3中可看出差别。

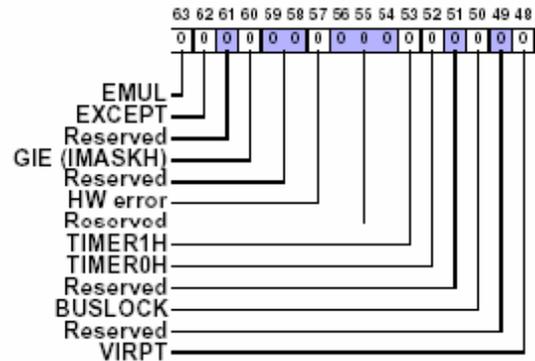


图2: IMASKH, ILATH, PMASKH (上部) 在ADSP-TS101S上的寄存器位

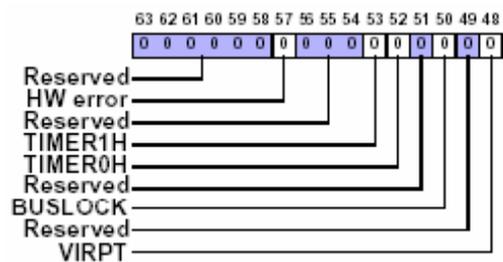


图3: IMASKH, ILATH, PMASKH (上部) 在ADSP-TS201S上的寄存器位

ILAT, IMASK 和PMASK寄存器的这些变化仅限于ILATH, IMASKH 和PMASKH (上部) 寄存器，所以只有部分ILAT, IMASK 和PMASK 寄存器在图中给出。ADSP-TS201S里的ILAT, IMASK 和PMASK寄存器的其它部分保持不变。

与ADSP-TS101S 相比，ADSP-TS201S上的定序器控制(SQCTL)寄存器里有一些位定义变化。

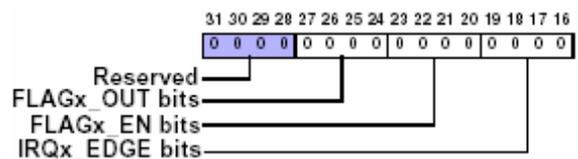


图4: SQCTL (上部) 在ADSP-TS101S上的寄存器位

位于ADSP-TS101S（图4）上的SQCTL寄存器内的27-20处的标志位，已移到ADSP-TS201S上的新FLAGREG寄存器上了（见图10）。

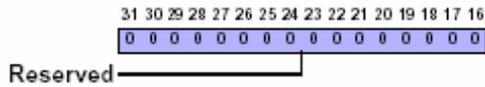


图5: SQCTL (上部) 在ADSP-TS201S上的寄存器位

位于ADSP-TS101S（图4）上的SQCTL寄存器内的19-16处的Edge/Level中断控制位，被移到了在ADSP-TS201S上的中断控制模块里的中断控制(INTCTL)寄存器处（见图11）。从图5可见，SQCTL寄存器内的27-16位被保留在ADSP-TS201S上。

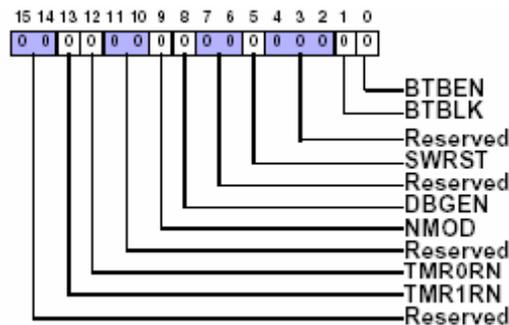


图6: ADSP-TS101S上SQCTL (下部) 寄存器位

位于ADSP-TS101S（图6）上的SQCTL寄存器内的13-12位的计时器控制位也被移到了ADSP-TS201S（图11）上的中断控制器内的INTCTL寄存器处。

异常和全局中断启用位（IMASK：是ADSP-TS101S内的62和60位，图2）已从IMASK寄存器中移出并加到ADSP-TS201S上的SQCTL寄存器（分别是图7里的3和2位）。仿真位（IMASK：ADSP-TS101S中的63位，图2）从IMASK寄存器中移出，EMUCTL寄存器里的

EMEN位包含此项功能。

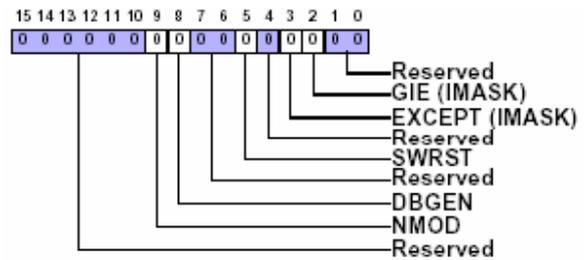


图7: ADSP-TS201S上SQCTL (下部) 寄存器位

位于ADSP-TS101S（图6）上的SQCTL寄存器内的1-0位处的BTB锁定（BTBLK）位和BTB启用（BTBEN）位已被移至ADSP-TS201S上的SQSTAT寄存器处，并由新的指令控制。

与ADSP-TS101S相比，ADSP-TS201S上的定序器状态(SQSTAT)寄存器的位定义也有一些变化。

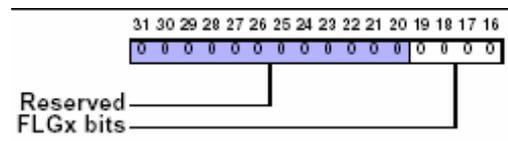


图8: ADSP-TS101S上的SQSTA (上部) 寄存器位。

从SQCTL寄存器（图6）里移出的BTBEN和BTBLK位分别位于ADSP-TS201S上的SQSTAT寄存器的28和29位（图9）处。启用BTB时，设置BTBEN。设置BTBLK时，所有新的登录都放入BTB并且设置锁定位。仿真，异常和HW中断位从PMASK寄存器移出，并被加到ADSP-TS201S（图9）上的SQSTST寄存器（分别是22，21和20位）处。

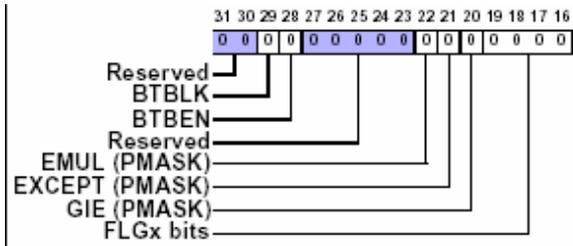


图 9：ADSP-TS201S 上的 SQSTA (上部) 寄存器位

为实现标志管脚的操作，定序器里增加了新的寄存器。图10 里的标志 (FLAGREG) 寄存器含从SQCTL 寄存器 (ADSP-TS101 中SQCTL 的 27-20 位) 中移出的标志方向位和标志值位。控制标志管脚 (3-0) 方向的位分别位于 3-0 位。表示标志管脚 (3-0) 的标志值位分别位于 7-4 位。注意：FLGx 位没有改变位置并且都位于 ADSP-TS101S 和 ADSPTS201S 上的 SQSTAT 寄存器内。FLAGREGST 寄存器用于在 FLAGREG 里设置位，而 FLAGREGCL 寄存器用于在 FLAGREG 里对位清零。

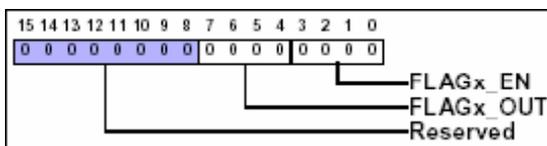


图 10：ADSP-TS201S 上的 FLAGREG (下部) 寄存器位

中断控制寄存器

ADSP-TS201S 上的中断控制器寄存器模块包含了很多从 ADSP-TS101 上的定序器处移出的定序器。移到中断控制模块的寄存器是 ILAT (低和高的，带设置和清除地址的)。IMASK (低和高的)，TIMER (0 和 1，低和高的) 和 TMRIN 寄存器 (0 和 1，低和高的)

ADSPTS201S 中的 LATL, IMASKL 和 PMASKL 寄存器的位定义与 ADSP-TS101S 相比没有变化，如前所述，除位被移到 SQCTL 和

SQSTAT 寄存器 (31: 28 位) 处的位以外，ILATH, IMASKH 和 PMASKH 寄存器的位定义与 ADSPTS101S 的相同。

ILATSTL 寄存器用于在 ILATL 里设置位，而 ILATCLL 寄存器用于在 ILATL 里清除位。类似的，ILATSTH 寄存器用于在 ILATH 里设置位，而 ILATCLH 寄存器用于在 ILATH 里清除位。

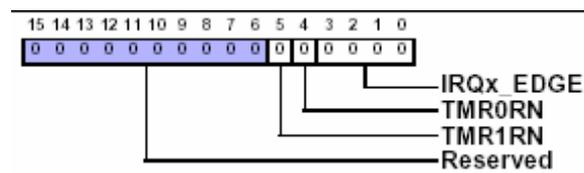


图 11：ADSP-TS201S 上的 INCTL (下部) 寄存器位

如前所述，ADSPTS201S 上的 INCTL 寄存器包含从 SQCTL 寄存器 (图 11) 移出的 Edge/Level 中断控制位和计时器控制位。IRQ3-0 的 Edge/Level 中断控制位分别位于 3-0 位。计时器控制位，TMR1RN 和 TMR0RN, 分别位于 5-4 位。INCTL 寄存器的其它剩余位被保留下来。

外部接口寄存器

与 ADSP-TS101S 相比，链接端口寄存器有一些变化。ADSP-TS201S 上的链接端口在被芯核和 DMA 访问方式上与 ADSP-TS101S 上的端口相似，但在物理实现上不同。图 12 和 13 强调了基本链接端口配置的差别。注意：ADSPTS201S 上的链接端口与前面的 TigerSHARC 处理器或 SHARC DSP 链接端口不兼容。

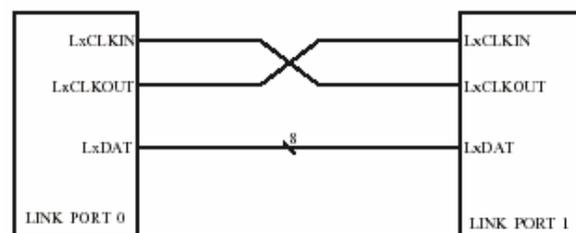


图 12：ADSP-TS101S 最小链接端口配置-无缓存

由于使用低压差分信号 (LVDS) 技术, ADSP-TS201S 的4个全双工链接端口的每一个都额外提供4比特的接收和4比特发送I/O的能力。由于具有在双倍数据速率 (即同时在500MHz频率时钟上升边和下边上锁定数据) 的运行能力, 每个链接端口都能提供每个方向上每秒500 Mb的支持, 使组合最大吞吐量达到4Gb/s。

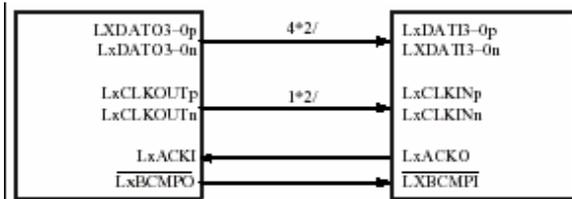


图13: ADSP-TS201S最小链接端口配置-终端未在图中给出

图13可见, ADSP-TS201S上的链接端口有了额外的管脚。至于ADSP-TS201S上的链接端口的硬件连接指南, 请参考EE-179 - ADSP-TS201S硬件系统设计指南。

链接缓存寄存器, LBUFTX_x, LBUFRX_x, (‘x’代表通道数3: 0) 如ADSP-TS101S所定义。新增加的寄存器用来反映链接端口的接收和发送能力, 即链接接收和链接发送控制寄存器(LRCTL_x, LTCTL_x), 链接接收状态及接收状态清零寄存器(LRSTAT_x, LRSTATC_x), 和发送状态及发送状态清零寄存器(LTSTAT_x, LTSTATC_x)。以上链接端口寄存器的位定义, 请参考defts201.h头文件。

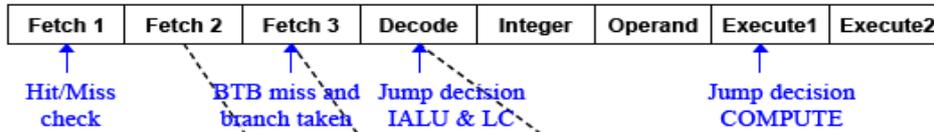
位功能寄存器变化概要

下表 (表1) 给出了寄存器从ADSP-TS101S到ADSP-TS201S的位定义变化的概要。

FUNCTION	ADSP-TS101S Register (BIT Locations)	ADSP-TS201S Register (BIT Locations)
FLAG _x _OUT	SQCTL - (27-24)	FLAGREG - (7-4)
FLAG _x _EN	SQCTL - (23-20)	FLAGREG - (3-0)
IRQ _x _EDGE	SQCTL - (19-16)	INTCTL - (3-0)
TMR _x RN	SQCTL - (13-12)	INTCTL - (5-4)
BTBLK	SQCTL - (1)	SQSTAT - (29)
BTBEN	SQCTL - (0)	SQSTAT - (28)
EMUL (IMASK)	IMASKH - (63)	EMEN bit in EMUCTL includes this functionality
EXCEPT (IMASK)	IMASKH - (62)	SQCTL - (3)
GIE (IMASK)	IMASKH - (60)	SQCTL - (2)
EMUL (PMASK)	PMASKH - (63)	SQSTAT - (22)
EXCEPT (PMASK)	PMASKH - (62)	SQSTAT - (21)
GIE (PMASK)	PMASKH - (60)	SQSTAT - (20)

表1: 寄存器位定义变化概要

ADSP-TS101S:



ADSP-TS201S:

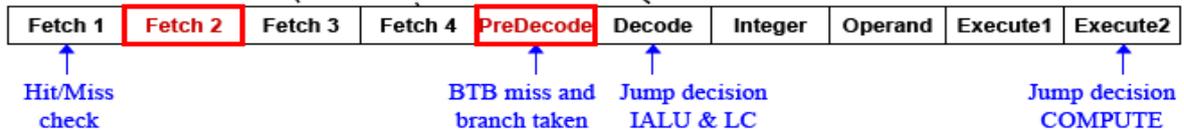


图14：程序指针流水线变化

程序指针

与ADSP-TS101S（见图14）相比，ADSP-TS201S上的程序指针有所变化。这些变化对用户来说是透明的，因为其对实际的源代码的兼容性无任何影响，由于两个新的流水线阶段而只对实际周期的计数有影响。

由于加入了两个新的流水线阶段，所以ADSP-TS201S上的程序定序器现在由一个10周期指令流水线组成，即一个4周期的取指流水线和一個6周期的执行流水线管道，而且计算结果在操作数出现后的两个周期就出来了。

相比于ADSP-TS101S 内的3个周期，ADSP-TS201S内的存储器访问需要4个周期。为适应更长时间的存储器访问，增加了另一个流水线阶段。ADSP-TS101S 里的取3阶段中进行的操作在ADSP-TS201S里是在取4阶段中进行的。

新增加的流水线阶段是预解码阶段，预解码阶段在取4阶段之后而在解码阶段之前发生。增加预解码阶段是为了给指令排列和指令解码更多的时间。

跟在ADSP-TS101S里一样，ADSP-TS201S支持分支预测，并包含了一个容许128个登录的分支

目标缓存（BTB）以缩短分支延迟来有效执行有条件的和无条件的分支指令，及零开销循环；正确预测与零开销周期一起发生的分支，克服 ADSP-TS101S 上的5-9 阶段的分支障碍。

注意：与ADSP-TS101S的情形一样，ADSP-TS201S的流水线也是完全互锁的，所以当依赖条件或资源冲突发生时，处理就自动插入延迟。延迟是上面提到的两个条件中的一个所引发的延迟。这对用户来说也是完全透明的，但由于这会降低性能，所以用户要尽量减少这些条件的发生。

内部存储器/高速缓冲存储器

与ADSP-TS101S 的6Mb内部DRAM存储器相比，ADSP-TS201S的内部存储器是总体容量为24Mb的内部DRAM存储器。ADSP-TS201S内部存储器分为6个单独的存储单元，每个存储器单元模块有4 Mb模块的内部DRAM的存储器空间，配置为128K x 32 比特字；如图15 所示，每个单元都有一个连接单元和扩展芯核总线的存储器交叉开关。

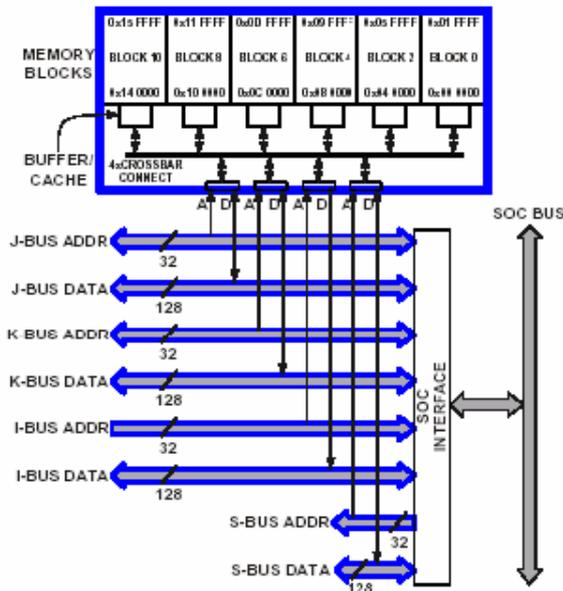


图15: 内部DRAM结构

每个4Mb的内部DRAM存储器块分为两个半块，半块0 和半块1，每个半块是由两个互相交织的1Mb子阵列构成，即子阵列0与子阵列1分别分别和子阵列2与子阵列3交织而成。一个子阵列分配成512个页面，每个页面2048比特，因而每页有8个256比特字。半块的子阵列是在页面基础上交织的。

了解更多ADSP-TS201S的内部存储器信息，请参考ADSPTS201S TigerSHARC 处理器编程参考手册的存储器和总线章节（第9章）。

访问内部存储器

如前所述，ADSP-TS201S内部存储器分为6个单独单元，总容量为24 Mb。每个单元包含一个存储器接口，一个内部DRAM的4Mb段和一个将单元连接至扩展芯核总线的存储器交叉开关（存储器块的物理结构请参见图16）。

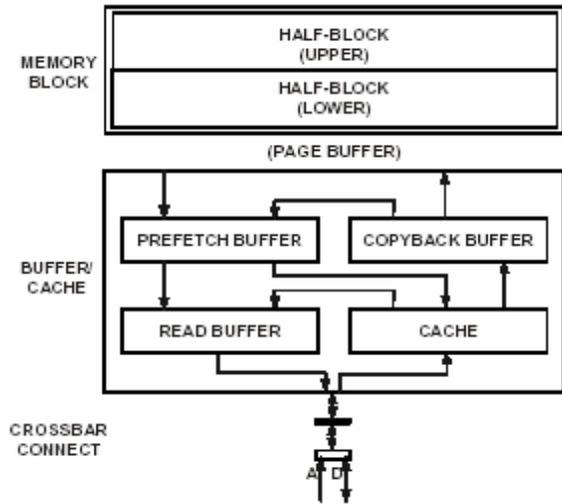


图16: 存储器块的物理结构

内部DRAM的时钟(DCLK)的运行频率是芯核时钟(CCLK)频率的1/2。为补偿较慢的时钟速度，内部DRAM使用的是比扩展芯核总线宽度一倍的基准字，即256比特。也要注意内部DRAM是单端口的。在每个存储器时钟周期里，它只能处理一个读或写操作。

与访问内部DRAM有关的障碍是反复无常的而且受多种因素的影响。了解这些障碍的来源对发展策略将对处理效率的影响降至最小是至关重要的。一个简单的，高水平的内部DRAM访问过程概要提供常规准则，即障碍何时发生，和如何避免其发生。

从内部DRAM上读出或往上写入都需要激活或打开适当的内部DRAM页面。激活页面的过程就是将数据页面从内部DRAM子阵列传输到相应的页面缓冲器上的过程。该过程只需要一个存储器时钟周期（两个芯核时钟周期）。

一旦页面被激活，一个256比特字就可读取或写入页面缓冲器。

一个“读”事务处理即是将一个256比特字从页

面缓冲器上传到预取缓冲器页面上的过程。类似地，一个“写”事务处理即是将一个256比特字从回拷缓冲器上传到页面缓冲器上的过程。该过程也只需要一个存储器时钟周期（两个芯核时钟周期）。

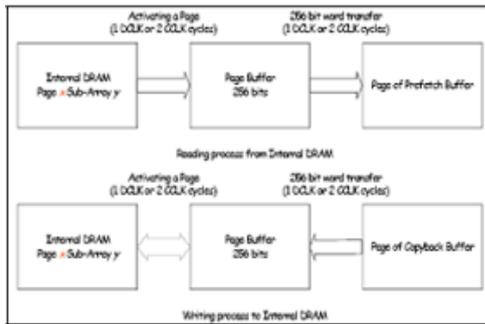


图17：内部DRAM读/写过程

最后，在相同子阵列里打开不同页面之前，页面必须关闭或预先充电。对页面预先充电就是将页面缓冲器上的数据页面传输回内部DRAM的过程。页面缓冲器的数据的任何改动都回在内部DRAM中反应出来。这个过程还是只需要一个存储器时钟周期（两个芯核时钟周期）。图17给出了内部DRAM读/写过程。

内部DRAM刷新

除了标准的读和写事务处理外，内部DRAM需要定期的刷新每个存储器页面。刷新页面简单地说就是激活一页然后对其预先充电的过程。这个过程是必需的，因为内部DRAM的基本存储器单元用电容器来存储数据。经过一段时间，电容会放电，这样存储器中的内容就会丢失。刷新过程定期地对电容器充电，并确保数据保持准确。用户开始编程前必须在芯核时钟里的CACMDALL功能中设置刷新周期CACMD_REFRESH。刷新周期显然是取决于处理器的芯核时钟频率的。当周期计数值达到设定值时，在不中断页面访问（同一页面里的页面预取和回拷序列在刷新前就已完成）的情况

下，应立即进行刷新。

注意：如果刷新与访问发生在同一存储器块里，就可能会有等待时间。

高速缓冲存储器模块

高速缓冲存储器模块由高速缓存数据单元和高速缓存控制单元组成。高速缓存控制单元对每个存储器事务处理的地址进行解码，并判断事务处理是为高速缓存命中还是高速缓存未命中。

如果事务处理是一个高速缓存命中，那么相应的高速缓存登录就无障碍地从高速缓存上读取或写入高速缓存。然而，如果事务处理是一个高速缓存未命中，那么就会招致障碍发生。障碍只是在数据必须从内部DRAM读取或写入时才发生。“读”事务处理上的高速缓存命中若为预取缓冲器命中或读缓冲器命中，就无需从内部DRAM读取，也就不会招致障碍发生。类似地，对并未强迫高速缓存回拷的写事务处理，高速缓存未命中也不需要写入内部DRAM，因而也不会招致障碍发生。

缓冲器模块

缓冲器模块，用于对从存储器读取或写入的数据进行排序以减少对内部DRAM的读写访问的频率。缓冲器模块包含3套完全不同的缓冲器：读缓冲器，预取缓冲器和回拷缓冲器。

512比特的读缓冲器由每个半块带一个256比特字的两个半块组成，它只用于读事务处理中，对所有读事务处理来说，它的功能是将存储器交叉开关和高速缓存与预取缓冲器连接到一起的唯一接口。注意：数据从未直接从内部DRAM上读取，而是在将数据移至读缓冲器前将数据拷在预取缓冲器上。该事务处理是写缓冲器命中，不会引发任何障碍。

8192比特的预取缓冲器由4页每页2048比特的页面构成（每页又分为8个256比特字），而且只用在读事务处理中。功能上它为从内部DRAM中读取的数据在传输到读缓冲器前提供一个媒介性质的存储场所。但是，预取缓冲器的首要功能是在真正事务处理请求前预测序列读事务处理和对来自内部DRAM的数据进行排列。预取缓冲器命中不引发任何障碍。

回拷缓冲器由两个单独的4096比特的双页缓冲器组成，每个双页缓冲器含16个256比特字为那些被替换的，最终要被写回内部DRAM的高速缓存提供一个媒介性质的保留场所。注意：对所有写事务处理，回拷缓冲器是内部DRAM的唯一接口。

与直接把数据写入内部DRAM相比，使用媒介式缓冲器是十分有益的。它提供了“写”事务处理对内部DRAM访问的延迟机制。延迟“写”事务处理访问，就能让待定的“读”事务处理无阻碍的完成。延迟的时段只受缓冲器的大小和缓冲器以多块速度被填满的限制。

存储器交叉开关

ADSP-TS101S有三个内部存储器块，每个都有2Mb的存储器空间。有三个单独的内部128比特数据总线，每个都与一个存储器快相连。在ADSP-TS101S上的一个周期内，在芯核中最多可发生3个128比特的传输（两个数据传输和一个程序指令传输。）

ADSP-TS201S处理器的扩展芯核包括：4个128-比特总线：S-Bus, J-Bus, K-Bus, 和 I-Bus。S-Bus用于协调SOC总线接口单元引发的内部存储器事务。J-Bus用于协调JALU引发的内部存储器事务。K-Bus用于协调KALU引发的内部存储器事务。I-Bus s用于协调定序器引发的内部存储器事务。所有扩展核心总线都在核心时钟频

率(CCLK)下运行。在最大的芯核时钟频率600MHz下，扩展芯核总线能支持38.4GB/s的组合（数据及指令）吞吐量。

存储器交叉开关的功能是作为桥来连接4个扩展芯核总线 and 单个的存储器单元。每个芯核时钟周期，它能支持4件事务，或更具体地说，一个总线周期里一件事务。事务的大小可以是32-比特字或64比特长字或128 比特四长字。注意：存储器交叉开关是内部存储器的唯一接口。

每个芯核时钟周期里，单个存储器单元只提供一件事务。当多个面向同一个存储器单元的事务在同一周期里启动时，就由一个裁决程序来解决冲突。事务优先级由高至低的顺序是：S-Bus事务，J-Bus事务，K-Bus事务，最后是I-Bus事务。由前个周期启动但被延迟的事务比在当前周期里启动的事务有更大的优先级。

访问内部存储器：编程指南概要

概括地说，内部DRAM的读/写过程是由高速缓冲存储器和缓冲器模块来促进完成的。这种结构的目的是防止用户程序在执行过程中被中断。在某些应用中为有效地利用内部存储器，这里推荐两种访问内部存储器的方法：顺序访问和局部访问。

顺序访问方法是由顺序地访问内部存储器构成。只有序列里的第一个访问是有障碍的，因为页面缓冲器是空的。首次访问后，访问就无任何障碍了。这种方法用在有大量存储器块，且同一数据只会有为数不多的访问的情形下十分有效。如果可能，用户不应为其它的访问而使用同一存储器区段。相反，用户应当将程序代码放在一个存储器的一个区段上而数据变量放在另一个区段内。同时，用户在同一个指令里也不要对同一（存储器）区段访问超过一次。

局部地址方法由对内部存储器的单个访问构

成，通常是必须要读或写数据的时候。因为如果页面缓冲器是空的，第一个访问通常是有障碍的，所以用户不应该试图使用顺序访问方法来访问内部存储器，或预先把数据载入高速缓冲存储器。该操作是通过在数据块上读/写，或通过利用DMA控制器在程序开始时完成的。或者可以锁定高速缓冲存储器，这样一些关键数据块可以为将来的访问预留下来。推荐在数据块比高速缓冲存储器(4K字)小的情形下使用这种方法。

DMA编程

DMA操作可以通过TigerSHARC处理器芯核，外部主机处理器，或(外部) TigerSHARC处理器总线主控来编程实现。DMA操作是通过写入存储器映射的DMA传输控制块(TCB)寄存器来编程实现的。每个TCB寄存器大小是128比特，分为4个每个32比特的寄存器。这四个寄存器是：索引寄存器(DI)，X维计数与增量寄存器(DX)，Y维计数与增量寄存器(DY)和控制与链指针(DP)。

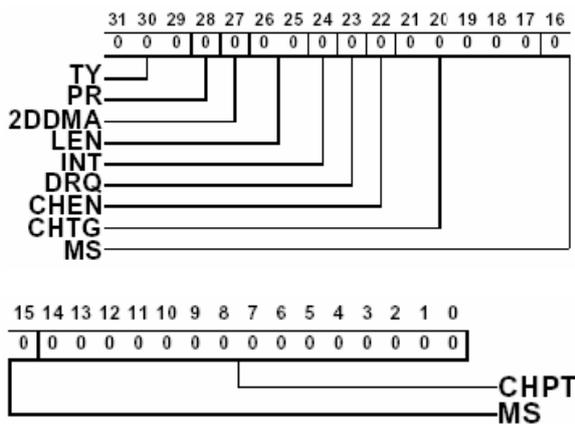


图 18：ADSP-TS101S上的DP 寄存器。

与ADSP-TS101S相比，ADSP-TS201S上的控制和链指针(DP)寄存器中的比特域有一些变

化。DP寄存器包含所有DMA的控制信息。该寄存器分为两个主字段。一个包含所有的控制信息，另一个包含所有的链信息。ADSP-TS101S上的DP 寄存器如图18。

ADSP-TS201S上的DP 寄存器如图19。注意：CHPT域包含位18-0(位15-0在图上没有给出)。

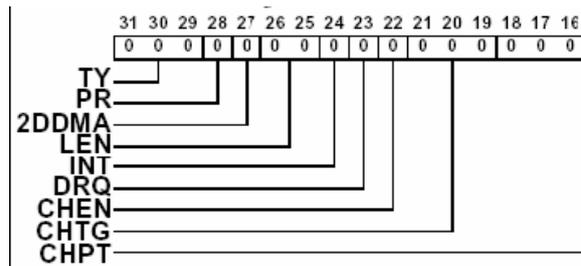


图 19：ADSP-TS201S上的DP 寄存器。

ADSP-TS101S上的DP寄存器内的MS(针对链指针的存储器选择)被移走，链指针比特域(CHPT)被扩大了以适应更大的存储器映射。链接目的通道(CHTG)比特域从ADSP-TS101S上的5比特(21-17)减至ADSP-TS201S上的3比特(21-19)。这就意味着一些原先在ADSP-TS101S上有效的链接通道在ADSP-TS201S上就不再有效了。

了解更多的DP寄存器和设置DMA传输的信息，请参考ADSPTS201S处理器硬件参考手册的直接存储器访问章节。

通讯逻辑单元

ADSP-TS201S 处理器芯核包括两个叫做计算块的计算单元。每个计算单元包括一个寄存器文件和其下的独立计算单元——一个ALU，一个乘法器和一个移位器。每个计算块也包括一个新单元：通讯逻辑单元(CLU)。

CLU包括面向通讯基础设施的ADSP-TS101S的

增强指令，用来控制管理Trellis解码（如Viterbi和Turbo解码器），通过复杂相关性和一个新操作 - 加入到ADSPTS201S上的新的交叉相关函数（XCORRS） - 来解除扩散。注意：这些函数也可以用于非通讯算法。图20突出显示CLU。

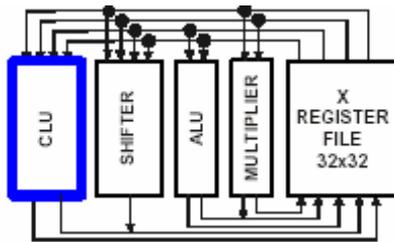


图20：计算块X中的CLU

CLU 从寄存器文件中提取它的输入，并将其输出也返回到寄存器文件中。大多数CLU指令操作在格架寄存器（TR）和格架历史寄存器（THR）上完成。

在TR寄存器文件（对应ADSP-TS101S里的16）和THR寄存器文件拓展为4个寄存器（对应ADSP-TS201S里的2）的情形下，加速器寄存器文件拓展为32个寄存器。

XCORRS指令将长输入序列（如一个8位引导器的2048个复数输入数）与已知的用于多次延迟的参考序列相关联。XCORRS指令可被看作16个并行DESPREAD指令的单循环执行。XCORRS指令的一些重要特征包括如下：

- 清零（CLR）选项，为格架寄存器在开始一个新的交联相关前提供清零机制。
- 删减输入（CUT）选项，为丢弃一些输入数据数（初始和结束周期的；高低三角的）提供了机制。
- 扩展精度（EXT）选项，支持16位输入和32位累加，而不是默认的8位输入和16位累加。

- 为每个延迟提供输出相关性强度。
- 存储器使用；执行时考虑存储器带宽和访问多块。

注意：通讯逻辑单元只存在于ADSP-TS201S上。ADSPTS202S和ADSP-TS203S上除了序列指令外，没有CLU指令。

指令集和指令功能性变化

下面描述指令集及其功能性的变化。

虚拟总线交换

寄存器之间或外部存储器与寄存器之间有交换发生时，交换可在任意一个内部总线上进行。这类交换称作虚拟总线交换，在第一个可用的内部总线上执行。

在ADSP-TS101S里，指令“Ureg = Ureg;”和“Ureg = <immediate value>;”使用了虚拟总线。因为只有一个虚拟总线仲裁器，所以在只有一种指令在单个指令线上执行。

ADSP-TS201S的内部总线结构是不同的。每个IALU都有自己的总线，而且只要它不将总线让与SOC接口，它就能在总线上执行任意的交换。在此结构里，这种类型的两个指令可并行执行，只要一个由JALU执行，另一个由KALU执行即可。

注意：由于KALU寄存器输出端口的争用，下列情形仍然是不合法的：

$[Jm + Jn] = Kx; Ureg = \langle \text{immediate} \rangle;;$

$[Jm + Jn] = Kx; Ks = Km + Kn \text{ (cjmp)};$

并发Ureg存储和更新

在ADSP-TS101S里，Ureg可能既不是总线交易

源，也不是同一周期内的指令结果。

如以下的结合在ADSP-TS101S上是不合法的。

```
xR0 = R1 + R2; [J0 + 4] = xR0;;
```

这种限制的原因是ADSP-TS101S里的仲裁会将交换延迟到指令结果都已写完的时候，并且写入存储器的数据是新的而不是旧的。但是在ADSP-TS201S里，无论总线何时可用，数据是应该在执行阶段E2传输到缓冲器（见图14）上的。这就消除了结果的不确定性，所以这在ADSP-TS201S里是合法的。

定序器寄存器和调试寄存器间的传输

定序器寄存器和调试寄存器间的传输限于ADSPTS201S上，也就是说，以下的交换在ADSP-TS201S上是不允许的：

```
<Debug register> = <Sequencer register>;
```

```
<Sequencer register> = <Debug register>;
```

```
<Debug register> = <Debug register>;
```

预测错误指令异常

在ADSP-TS101S里，即使条件是错的并且指令并没有被真正执行的时候，预测指令也能引发异常。在ADSP-TS201S里，预测错误指令不能引发异常。

空闲指令

当在ADSPTS201S上使用空闲指令时，选项（LP）已被移出。出于兼容性上的考虑，ADSP-TS101S上的下列指令将定期执行ADSPTS201S上的空闲指令：

```
IDLE (LP);;
```

SOC 器件Ureg 传输异常

SOC 器件Ureg 不可能被转移到定序器或调试寄存器。如下列指令，在ADSP-TS101S上是合法的指令，而在ADSP-TS201S上却是不合法的：

```
WPOCTL = LOSTAT;;
```

解扩散指令

在ADSP-TS201S上，解扩散的格式也有所变化。新的格式是：

```
TRs += DESPREAD (Rmq, THrd);;
```

ADSP-TS101S里的形式是：

```
TRs = DESPREAD (Rmq, THrd) + TRn;;
```

这个变化是ADSP-TS201S中，累加必须是在同一个寄存器里（也就是说在ADSP-TS101S里，TRn 可以与 TRs 不同，而在ADSP-TS201S里，它们必须是相同的。）注意，如果TRs = TRn，汇编程序允许旧的（ADSP-TS101S）格式。

ADSP-TS20xS 系列产品差异

ADSP-TS20xS 系列中的三个处理器的差异是基于时钟速率，存储器大小，链接端口数目，链接端口带宽和外部端口宽度的。ADSP-TS20xS 处理器间的差异在下面的表 2 中予以强调。

ADSP-TS201S 运行的最大时钟频率为 600 MHz，有 24 Mb 的片上存储器，为通讯应用优化的通讯逻辑单元（CLU）的指令，有 4 个可在 CLK 频率下运行的链接端口。

ADSP-TS202S 运行的最大时钟频率为 500 MHz，有 12 Mb 的片上存储器，无 CLU 指令，有 4 个可在 CLK 频率下运行的链接端口。

DSP-TS203S 运行的最大时钟频率为 500 MHz，有 4 Mb 的片上存储器，只有 2 个可在

1/2的CCLK频率下运行的链接端口，无CLU指令。

ADSP-TS203S处理器的外部端口不支持64位的操作，所以SYSCON寄存器内的外部总线带宽控制位（位21-19）必须=0，其它任何值在ADSP-TS203S处理器都是非法的。

因为只有两个链接端口，并且在ADSP-TS203S处理器上外部端口限于32位。ADSP-TS203S处理器和其它ADSP-TS20xS处理器间有有很多管脚差异。请参照ADSP-TS20xS系列每部分管脚的数据表。

特征	ADSP-TS201	ADSP-TS202	ADSP-TS203S
最大核心时钟速率	500/600 MHz	500MHz	500MHz
片上存储器	24Mb 内部 DRAM	12Mb 内部 DRAM	4Mb 内部 DRAM
通讯逻辑单元	有	无	无

参考文献:

- [1] ADSP-TS101S TigerSHARC Processor Hardware Reference, Rev. 1.0 (06/03), Analog Devices, Inc.
- [2] ADSP-TS201S TigerSHARC Processor Hardware Reference, Rev. 0.1 (07/03), Analog Devices, Inc.
- [3] ADSP-TS101S TigerSHARC Processor Programming Reference Rev. 1.0 (01/03), Analog Devices, Inc.
- [4] ADSP-TS201S TigerSHARC Processor Programming Reference, Rev. 0.1 (06/03), Analog Devices, Inc.
- [5] ADSP-TS101S TigerSHARC Embedded Processor Datasheet. Rev. A (03/03), Analog Devices,

链接端口	4个：总计4 GB/s	4个：总计4 GB/s	2个：总计4个1 GB/s
外部端口	64 / 32位操作	64 / 32位操作	仅32位操作

表2：ADSP-TS20xS系列产品差异

结论

ADSP-TS201S与ADSP-TS101S间存在很多差别，但是两种处理器的芯核根本上讲是一样的。将ADSP-TS101S上的应用移到ADSP-TS201S上在代码上需要有所变化，但会有速度，存储器大小，外部端口，连接端口等的优势。本工程师对话简单讨论了将ADSPTS101S应用上的代码移到ADSP-TS201S上所要考虑的一些问题。对ADSP-TS20xS系列内部的差异也进行了讨论。

Inc.

[6] ADSP-TS201S TigerSHARC Embedded Datasheet. Rev. PrG (06/03), Analog Devices, Inc.

[7] ADSP-TS202S TigerSHARC Embedded Datasheet. Rev. PrA (06/03), Analog Devices, Inc.

[8] ADSP-TS203S TigerSHARC Embedded Datasheet. Rev. PrA (06/03), Analog Devices, Inc.

[9] ADSP-TS201S Hardware System Design Guidelines (EE-179)

文件历史

版本	描述
2003年9月3日, S. Francis供稿	第一版