

文章编号 1004-924X(2009)12-3096-07

TMS320C6416T 的 MPEG-4 编码优化

李桂菊, 陈秋萍, 纪 华, 武治国, 刘艳滢

(中国科学院 长春光学精密机械与物理研究所, 吉林 长春 130033)

摘要: MPEG-4 是当前最具影响的多媒体数据压缩编码国际标准。针对视频编码过程中计算复杂度高, 系统运算量大等特点, 本文提出了在 TMS320C6416T 上实现 MPEG-4 编码的优化方法。为使 MPEG-4 编码器达到实时应用的要求, 结合 C64x DSP 的特点, 采取了一系列软件优化方法, 包括合理分配存储器, 在内存开辟双缓冲区结构, 用汇编程序实现 MPEG-4 的核心算法以及对整个程序流程进行 C 语言级优化等。实验结果表明, 该编码器可以对 768 pixel×576 pixel, 25 frame/s 的视频图像进行实时编码。

关键词: 视频压缩; MPEG-4 标准; TMS320C6416T; 软件优化

中图分类号: TN919.81 **文献标识码:** A

Optimization of MPEG-4 video encoder based on TMS320C6416T

LI Gui-ju, CHEN Qiu-ping, JI Hua, WU Zhi-guo, LIU Yan-ying

(Changchun Institute of Optics, Fine Mechanics and Physics,
Chinese Academy of Sciences, Changchun 130033, China)

Abstract: MPEG-4 is one of the most important and influential international multimedia data compression encoders. To meet the demands of a video encoding for the large computation load and strong computing power, a optimized encoding method of MPEG-4 based on TMS320C6416T is presented. According to the characteristics of C64x DSP, several kinds of methods for software optimization are proposed, which includes allocating memories in reason, setting two buffers in a memory based on the characteristic of EDMA, realizing a core code by linear assembly language and optimizing C language for whole programs. Experimental results show that the encoder can realize real-time encoding for video images with 768 pixel×576 pixel, in 25 frame/s.

Key words: video compression; MPEG-4 standard; TMS320C6416T; software optimization

1 引 言

MPEG-4 是当前最重要最具有影响的多媒体数据压缩编码国际标准。随着信息技术的发展,

视频编解码过程中的计算复杂度越来越高, 数据运算量也越来越大, 因此要求系统具有很高的数据吞吐量和实时性^[1-3]。当前, 各种高性能 DSP 的出现为高效媒体处理器的实现提供了可靠的硬件平台。如何将 MPEG-4 在 DSP 上实时实现已

收稿日期: 2008-11-07; 修订日期: 2008-12-23.

基金项目: 国家 863 高技术研究发展计划资助项目 (No. 2006AA703405F)

成为视频编码领域的研究热点^[3-6]。文献[4]和文献[7]介绍了在 TMS320DM642 DSP 平台上实现视频编码的 MPEG-4 视频编码器。DM642 虽然以其包括集成的多媒体与通信外设受到许多图像压缩者的青睐,但其内存较小,片内存储容量仅为 288 kB,使得程序和数据都需要频繁地与外部存储器进行交换,因此不利于整个系统性能的提高。文献[5]采用 TI 公司开发的浮点 DSP 芯片 TMS320C67XX,在主频 150 MHz 的情况下,定点运算仅为 1 200 MIPS,浮点运算速度为 900 MFLOPS,与定点 DSP 芯片 TMS320C6416T 8000 MIPS 相比速度较慢,而 MPEG-4 中的关键算法又都可由定点完成,因此用浮点芯片做压缩发挥不出其优越性。基于上述工作的特点,本文提出在 TI 公司的 TM320C6416T 平台上进行压缩编码处理。TMS320C6416T 片内存储容量为 1056 kB,全速运行时能够以 8000 MIPS 速度处理信息。在实现过程中对 MPEG-4 开源代码 Xvid 1.1.0 做了简化,对 I-VOP、P-VOP 分别进行编码,在此基础上进行了有关程序结构、数据结构和算法的改进,以便充分发挥 DSP 的性能;并进行了 C 语言级和汇编语言级的优化。

2 存储器配置

TMS320C6416T 片内采用 2 级存储器结构^[8-9],第一级存储器包括相互独立的 16 kB 程序 cache(L1P)和 16 kB 数据 cache(L1D),L1 距离 DSP 核最近,数据访问速度最快,但只能作为不能寻址的高速缓存被 CPU 访问。第 2 级存储器 L2 是一个统一的程序/数据空间,容量较大,有 1 024kB,可以总体作为 SRAM 映射到存储器空间,也可以整体作为第 2 级 cache,或者作为二者按比例的一种组合混合使用^[5]。如果能恰当运用 C6416T 的两级缓存结构,将能极大地提高程序性能。在实际应用中改变 L2 的配置模式,对运行速度影响很小,因此将 L2 配置为纯 SRAM 模式,即全部作为 TMS320C6416T 片上内存,地址空间从 0 到 0x0ffff。

开发程序时要充分考虑存储器的特性,从尽量节省存储空间、提高代码在缓存的命中率两方面着手,修改程序结构和数据结构。

(1) 在定义变量时将 char, short, int, float, double 等同类型尽量放在一起。

这是由 c6000 变量存放机制所决定的。C6000 系列 DSP 的 char 型是 8 bit 可以放在内存的任意位置,short 型变量是 16 bit,占 2 Byte,是以 2 的整数倍为边界存放,int 和 float 型变量是 32 bit,占 4 Byte,是以 4 的整数倍为边界存放,double 型变量是 64 bit,占 8 Byte,是以 8 的整数倍为边界存放。例如在 PC 机上,变长编码表 VLC 是定义成 8 bit 长度变量,和 32 bit 的码值变量的数据结构:

```
typedef struct VLC
{
    uint32_t code;
    uint8_t len;
}
```

对于 $64 \times 64 \times 2 \times 2 = 16\ 384$ 大小的 DCT 系数查找表占用 $16\ 384 \times 8$ bit,在 8 bit 变量之后,紧接着存放 int 型变量,跳过 3 Byte,找到 4 的整数倍边界。若不定义结构,而是把长度和码值单独存放只需 $16\ 384 \times 5$ bit,节省 3/8 空间。

(2) 对于较大数组还要考虑是否能用 short 型代替 int 型,如在 PC 机上运动向量 (x, y) 定义成整型,而其值的范围在 16 bit 之内,故改用 short 型。

(3) 对于灰度图像的压缩,可以取消彩色编码表,为了兼容解码程序,在编码中认为宏块中 U、V 分量为零,经过变量存放优化后节省了大量空间,变量及各种表格仅占用了 92 kB 存储空间。

(4) L2 的分配。考虑要放到 SRAM 的是表格,全局变量,栈数据和一些调用频繁的核心程序,如运动搜索,DCT 变换,量化而整个当前图像和参考图像就只能放在片外存储空间了。存储器分配如图 1 所示。

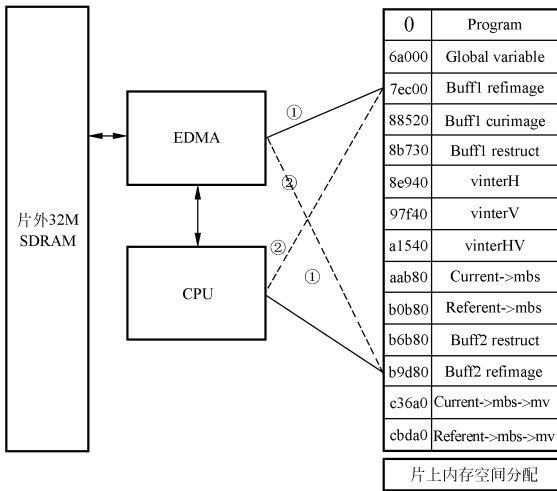


图 1 双缓冲区结构和片上内存分配

Fig. 1 Dual_buffer structure and RAM allocation

3 EDMA 图像数据搬移

把当前图像和参考图像数据都放在片外存储器,就要涉及到数据在片内存储跟片外存储间的数据搬移^[11],这可交由 C6416 强大的 EDMA 引擎来完成。EDMA 工作时不占用 CPU 的周期,把 CPU 从繁重的搬移数据的工作中解放出来,专致于运算工作。在编码程序时,为了避免 CPU 等待 EDMA 搬完数据后才能工作,可采用双缓冲区结构^[12]。对 I 帧编码,仅需分配两个 16 行大小的图像区域。对 P 帧编码,除分配一个 16 行大小的当前数据存储区,每个缓冲区还需分配 16 行重构图像和包括搜索范围内的参考帧图像存储区。如图 1 所示。当 EDMA 传送数据到其中一块存储区域时,CPU 对另一块存储区域进行处理,待二者都处理完毕后,缓冲区指针交换。这样双缓冲区交替使用,处理器和 EDMA 都能高效运作,减少了 CPU 的等待时间。EDMA 在搬大量数据时才能将它的性能发挥到极致,如果每编完一个宏块就进行一次缓冲区交换,那么在频繁的配置 EDMA 通道参数上就耗费了过多的 CPU 周期。有限的片内存储空间,制约着不能一次搬太多的宏块,故我们设置一次搬移 16 行图像数据,即 48 宏块。

为了提高 EDMA 的效率,可以采用 EDMA 链,一次开辟多个 EDMA 通道,让其首尾相连,这

样只需触发一次 CPU,可将待编码的一条图像数据一次搬完。在配置 EDMA 通道时,频繁更换的只是 EDMA 的源地址和目的地址,而其它参量是不变的,故可以直接在相应的配置表上写上新地址,而不必调用 CSL 库中的相应的 cache 函数来修改源、目的地址。需要注意的是,EDMA 的数据通道只有一条,故在开一个 EDMA 时,为保证读写数据正确,必须判断其他开启的 EDMA 是否已经完成工作。

DSP 对不同存储区的访问速度不同,对片外存储器的访问速度为 200 MB/s,而对片上访问速度可达到 8 GB/s,鉴于如此悬殊的速度差异,我们把程序做如下改进:

(1) 把一幅图像的宏块运动信息存储在片上内存,避免由于片内外宏块信息搬移造成 CPU 周期的耗费和 EDMA 通道的等待;

(2) 一条图像编码结束后把一条参考图像 (800 pixel \times 49 pixel) 的插值结果的后 32 行往上搬移 17 行,则对下一条参考图像的插值可以只计算最后 17 行。

4 算法优化

直接移植到 DSP 上的 MPEG-4 视频编码器编码效率低,远远不能达到实时的要求,因此还需要结合 TMS320C6416T 的特点进行优化。编码器的计算量主要集中在运动估计、量化、反量化、DCT、IDCT 和可变长编码几个模块。算法支持半像素运动估计,这样就需要进行半像素插值处理,包括行插值、列插值和对角线插值,这个过程不复杂但是运算量比较大。为了减少运算量,对这些模块进行了不同的优化处理。首先合并一些模块并编制了汇编算法,包括:

(1) 插值算法:同时对一条图像进行行插值、列插值和对角线插值;

(2) 量化变换算法:该算法对一个 8 \times 8 的块执行,包括从 8 bit 图像中读数据,并转换成 16 bit,进行 DCT 变换、量化,得到量化系数;

(3) 逆变换逆量化算法:该算法对一个 8 \times 8 的块执行,包括读量化系数,进行逆量化,逆 DCT 变换,变换后的 16 bit 数据转换成 8 bit,并存到 8 bit 重构图像中。

其次将调用次数较多的函数改写成汇编,如

码流输出中的 BitstreamPutBits 和 BitstreamPutBit 函数,运动估计中的匹配位置搜索函数等。

为了得到优化性能最佳的代码在对编写并行汇编及调用 c64x 系列的专用库函数外,还对整个工程进行了 PBC 优化。

5 程序结构优化

本文的 MPEG-4 算法模型以条 (16 pixel × 768 pixel) 为单位进行片内外数据搬移、参考图像插值、运动估计,以宏块为单位进行运动补偿、纹理编码和运动信息编码,这样可以增强程序和数据的局部性,提高程序和数据 cache 的命中率。

表 1 优化后各个模块的处理时间

Tab. 1 Executive time of module after optimization

模块	运算介质	CPU 周期数
读一条当前图像	EDMA	65 889
读一条重构图像	EDMA	201 956
写一条重构图像	EDMA	66 308
对一条参考图像插值包括搬移	CPU	71 494
一条图像运动估计	CPU	>228 993
P 帧一条图像纹理编码	CPU	>108 468
I 帧一条图像纹理编码	CPU	>949 956

表 1 是相机随机移动测到的时间。根据上面测得的模块处理时间表,利用 EDMA 的数据搬移不占用 CPU 时间的特性,我们设计了 CPU 流程和 EDMA 流程,如图 2、图 3 所示。在对一条参考图像插值前,开启 EDMA 完成一条当前图像的读取、下一条参考图像(49 pixel × 768 pixel)的读取和上一条重构图像到外部 SDRAM 的写操作,其中三个 EDMA 通道利用链接方式完成相继启动。在完成一条参考图像插值时,我们由上面的时间表可以知道,一条当前图像的读取必然已经完成,故可以把这条当前图像移到重构图像地址,继而完成后面相应的编码操作。在一条图像编码结束后,交换两个 Buff 指针,循环上述操作。在一帧图像编码结束后开启 EDMA,完成下一帧第一条参考图像的读取,这样在进入下一帧处理时,可以直接对第一条参考图像进行插值,让 CPU 高效运转,提高图像编码速度。优化后的 I、P 帧

编码流程图如图 2、3 所示。

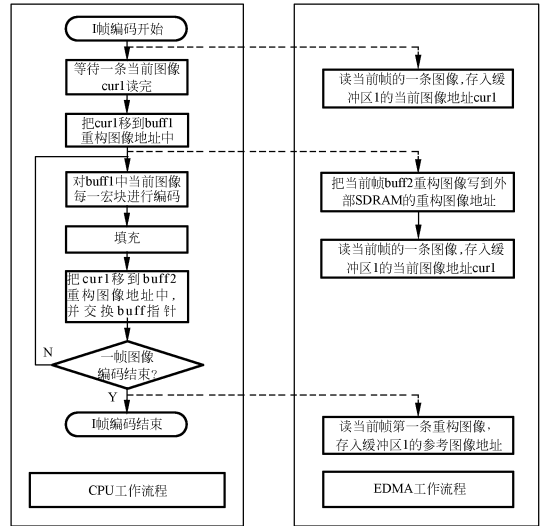


图 2 I-VOP 的编码流程

Fig. 2 Flow chart of I-VOP encoding

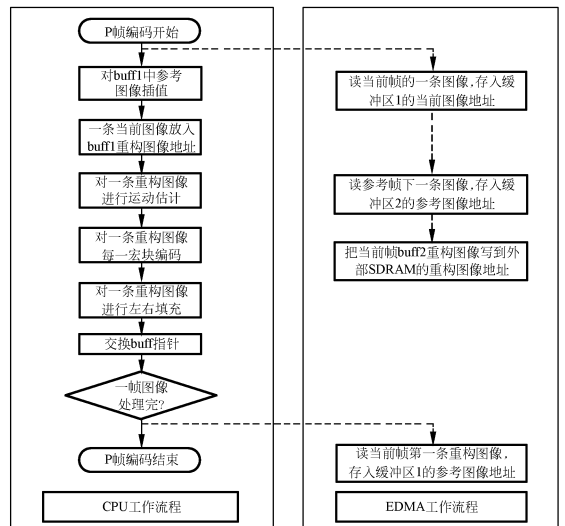


图 3 P-VOP 的编码流程

Fig. 3 Flowchart of P-VOP encoding

由上分析可知,整个数据的搬移过程基本不占 CPU 的时间,加快了一帧图像的编码速度。

6 实验结果分析

下图给出了两组序列图片。第一个序列是从 Google 下载的机场图片,在仿真环境下,首先把图片序列存到 DSP 的外部存储器中,统计时去掉读写图片的时间。第二个序列是相机实时采集的图片。

表 2 给出了优化前后主要模块时间对比。不同的图像序列,时间差异较大,表中 I 帧时间是

图 4 第一帧的平均时间,P 帧时间是图 4 的第 25 帧平均时间。

表 2 MPEG-4 编码器优化前后主要模块时间对比

Tab. 2 Executive time comparison of main module in MPEG-4 Encoder before and after optimization

	优化前/ μs	优化后/ μs	性能提高倍数 (优化前/后)
I 帧量化变换算法(一个宏块)	58.2	4.47	13.02
I 帧逆变换逆量化算法(一个宏块)	65.7	9.2	7.14
P 帧量化变换算法(一个宏块)	30.1	2.3	13.09
P 帧逆变换逆量化算法(一个宏块)	31.4	5.1	6.15
运动估计(16×768)	1 487.5	217.6	6.84
插值算法(16×768)	819.8	46.02	17.8

表 3 MPEG-4 编码器优化前后性能比较

Tab. 3 Performance comparison of MPEG-4 Encoder before and after optimization

	优化前		优化后	
	PSNR	帧频/(frame/s)	PSNR	帧频/(frame/s)
图片系列 1	36.38	5.1	36.38	40.8
图片系列 2	34.8	3.9	34.7	42.2

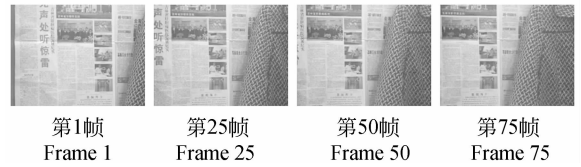


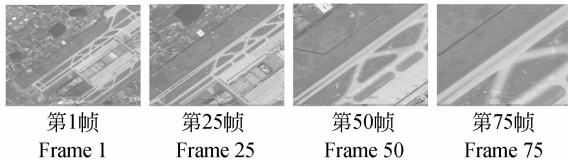
图 5 实时采集的图像序列(压缩后图像)

Fig. 5 Real-time images (after compression)

表 3 给出 2 个图片系列压缩的信噪比和帧频,可以看出本文设计的 MPEG-4 编码器能够实现 40 frame/s 左右的编码速度,可以满足实时编码的要求。

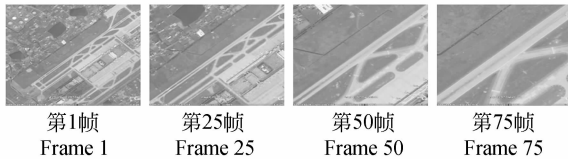
7 结 论

本文介绍了图像压缩算法在专门设计的基于 TMS320C6416T 的硬件平台上的实现及优化。通过合理分配存储器、使用 EDMA、调用 TI 提供的库函数和对主要模块编写并行汇编等软件优化方法,使该平台具有较高的数据吞吐能力和处理速度,能够对 768 pixel \times 576 pixel 大小的图像进行实时压缩,并已应用在实际工程中。



(a) 压缩前图像

(a) Images before compression



(b) 压缩后图像

(b) Images after compression

图 4 Google 下载的机场图片

Fig. 4 Airplane images download from Google

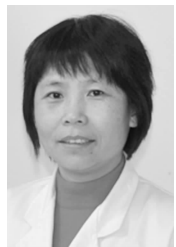
参考文献:

[1] 钟玉琢,王琪,贺玉文. 基于对象的多媒体数据压缩编码国际标准-MPEG-4 及其校验模型[M]. 北京:科学出版社,2000.
ZHONG Y ZH, WANG Q, HE Y W. *Based on Video Object Multimedia Data Compress Encode*

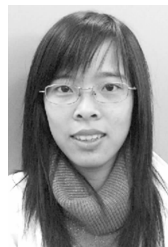
International Standard-MPEG-4 and Video Verification Model [M]. Beijing: Press of Science, 2000. (in Chinese)
[2] 张石,张明亮,鲍喜荣,等. MPEG-4 视频解码模块的设计与优化[J]. 计算机工程,2007,33(10):193-195.
ZHANG SH, ZHANG M L, BAO X R, et al.. *De-*

- sign and optimization of MPEG-4 video decoder [J]. *Computer Engineering*, 2007, 33(10): 193-195. (in Chinese)
- [3] GUO H, SHENG T, SUN W. Cache optimization for an embedded MPEG-4 video decoder[C]. *The 8th International Conference on signal Processing, Nanning*, 2006: 16-20.
- [4] 周雅贇, 徐元欣, 方健, 等. 基于 TMS320DM642 的 MPEG-4 编码器设计和优化[J]. *电视技术*, 2005, 29(6): 36-38.
ZHOU Y Y, XU Y X, FANG J, et al.. Implementation of mpeg-4 video encoder based on TMS320DM642[J]. *TV Engineering*, 2005, 29(6): 36-38. (in Chinese)
- [5] 曾明霞. 基于 DSP 的 MPEG-4 视频编码器研究与实现[D]. 南京: 南京理工大学, 2007.
ZENG M X. *Research and implementation of MPEG-4 video coder based on the the DSP* [D]. Nanjing: Nanjing University of Science and Technology, 2007. (in Chinese)
- [6] 赵峰, 袁东风, 张海霞, 等. 多 DSP 图像压缩实时并行处理系统[J]. *光学精密工程*, 2007, 15(9): 1451-1455.
ZHAO F, YUAN D F, ZHANG H X, et al.. Multi-DSP real-time parallel processing system for image compression[J]. *Opt. Precision Eng.*, 2007, 15(9): 1451-1455. (in Chinese)
- [7] 费伟, 朱向军, 裘赛海, 等. 基于 C64x DSP 的 MPEG-4 视频编码器算法设计及优化[J]. *信号处理*, 2007, 23(2): 256-261.
FEI W, ZHU X J, QIU S H. Algorithms design and optimization of MPEG-4 video encoder using C64x DSP [J]. *Signal Processing*, 2007, 23(2): 256-261. (in Chinese)
- [8] Texas Instruments. TMS320c64x DSP Two-level Internal Memory Reference Guide[R]. *SPRU610b, Texas Instruments*, 2004. 8.
- [9] Texas Instruments. TMS320C6000 DSP Cache User's Guide [R]. *SPRU656A, Texas Instruments*, 2003. 05.
- [10] 李方慧, 王飞, 何佩琨. TMS320C6000 系列 DSPs 原理与应用[M]. 2 版. 北京: 电子工业出版社, 2003.
LI F H, WANG F, HE P K. *DSPs Principle and Applying of TMS320C6000* [M]. 2nd ed. Beijing: Electronics Industry press, 2003. (in Chinese)
- [11] 王刚, 王世刚. 基于 TMS320DM642 DSP 的 MPEG-4 视频编码器优化[J]. *吉林大学学报(信息科学版)*, 2006, 24(6): 590-593.
WANG G, WANG SH G. MPEG-4 video encoder realization and optimization based on TMS320DM642 DSP [J]. *Journal of Jilin University (Information Science Edition)*, 2006, 24(6): 590-593. (in Chinese)
- [12] 陈升来, 黄廉卿, 郭静寰. 基于整形提升小波变换的图像处理及 DSP 实现[J]. *光学精密工程*, 2006, 14(3): 498-502.
CHEN SH L, HUANG L Q, GUO J H. Image processing based on integral lifting scheme and its implementation by DSP [J]. *Opt. Precision Eng.*, 2006, 14(3): 498-502. (in Chinese)

作者简介:



李桂菊(1964—),女,吉林省吉林市人,研究员,1985年于吉林大学获得学士学位,1988年于中科院长春光学精密机械与物理研究所获得硕士学位,主要从事数字图像处理及 DSP 应用方面的研究。E-mail: lgjciom666@ yahoo.com.cn



陈秋萍(1985—),女,福建莆田人,硕士研究生,主要从事图像压缩方面的研究。E-mail: 305728033@qq.com

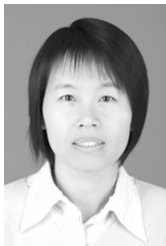


纪 华(1983—),女,河北唐山人,博士研究生,主要从事景象匹配、视频跟踪方面的研究。E-mail: jhua12@163.com



武治国(1978—),男,吉林长春人,博士研究生,助理研究员,主要研究方向为图像处理和数据融合。E-mail: wu78zg@163.com

通讯作者:



刘艳滢(1965—),女,辽宁营口人,研究员,1988年于湖南大学获得学士学位,主要从事数字图像处理的研究。E-mail: liuyy@ciomp.ac.cn

● 下期预告

Nd:YAG 激光制备连杆初始裂解槽试验结果及分析

郑祺峰,杨慎华,林宝君,寇淑清
(吉林大学 锻锻研究所,吉林 长春 130025)

裂解槽的设计与加工是连杆裂解加工技术的核心和技术关键,将直接影响着连杆的裂解质量。为改善连杆裂解槽加工精度和质量,进而提高裂解连杆的成品率,通过两种连杆产品试制,确定激光加工参数。首先采用 Nd:YAG 固体激光器对两种类型的连杆进行了裂解槽激光加工试验,而后对不同切割参数加工的裂解槽质量加以分析,并对激光切割参数进行了优化。结果表明:激光峰值功率、离焦量、切割速度、脉冲频率、辅助气体压力、激光入射角等对裂解槽的加工质量均有很大的影响。其中激光峰值功率 2.4 kW,脉冲时间 0.4 ms 时,切槽深度为 0.453~0.457 mm,当脉冲频率与切割速度比值约为 3,并在非正离焦量状态下时,所加工的裂解槽对连杆的裂解都非常有利,能够满足 M0406 连杆的裂解质量要求。