

i486 处理机

HOPE

程序员参考手册

(共二册)

(上册)

鹏飞 晓晖 译 博 山 校

北京希望电脑公司



i 4 8 6 处 理 机

程序员参考手册

(上 册)

鹏飞 晓晖 译 博 山 校

北京希望高级电脑技术公司

一九九一年八月

i 4 8 6 处 理 机

程序员参考手册

(下 册)

鹏飞 晓晖 译 博 山 校

北京希望高级电脑技术公司

一九九一年八月

前 言

i486 处理器为 DOS、OS/2、Windows 和 UNIX 系统 V/386 应用程序提供了高性能的服务。它具有由一百万个晶体管集成的高速缓冲存储器、浮点硬件和内存管理功能,并保留了与其它 86 系列处理器的兼容性。规整的 8KB 编码方式和数据缓冲,加上主频 25/33MHz、每秒 80/106MB 的总线,保证了系统在仅配有廉价的 DRAM 时,也具有很高的吞吐量。

i486 处理器的特性包括:

- ☐ 与 386DX/SX、376、80286、8086 和 8088 处理器完全二进制兼容,与 386 DX/SX、376 处理器目标代码级兼容。
- ☐ 使用频度高的指令可以在一个时钟周期内完成,达到了 RISC 性能级。
- ☐ 用 32 位的整数处理器执行算术和逻辑运算。
- ☐ 浮点算术部件支持 IEEE 754 标准所指定的 32、64 和 80 位格式(与 387DX/SX 协处理器目标代码级兼容)。
- ☐ 8KB 缓冲存储器可以快速存取最近使用的指令和数据。
- ☐ 总线控制信号可以在多处理器系统中维护缓冲的一致性。
- ☐ 分段的存储管理机构可建立独立的、受保护的地址空间。
- ☐ 分页的存储管理机构支持利用硬盘作为虚拟内存。
- ☐ 重启指令允许程序在异常发生后重新开始执行(支持页面申请方式的虚拟存储器时必需)。
- ☐ 管道指令的执行可同时解释不同的指令。
- ☐ 调试寄存器提供了对指令和数据断点的硬件支持。

i486 的操作方式决定了指令集和结构特性。i486 处理器有三种运行方式:保护方式、实地址方式和虚拟 8086 方式。

《i486 程序员参考手册》分为上、下两册,包括以下五个部分:

- | | |
|---------------------------------------|------------------------------------|
| ☐ 第一部分——应用程序设计 | ☐ 第四部分——兼容性 |
| ☐ 第二部分——系统程序设计 | ☐ 第五部分——指令集 |
| ☐ 第三部分——算术处理 | ☐ 附录部分 |

上册详尽讲述了基本程序设计模式、应用程序指令集、系统体系结构、存储管理、保护、多任务、输入/输出、初始化、调试、高速缓存、多机处理、数值应用程序、浮点算术部件、特殊的计算情形、浮点指令、算术例程、运行 80286 和 386DX 程序、实地址方式、虚拟 8086 方式、混和 16 位和 32 位代码,以及与 8087/80287 和 387DX 协处理器的兼容性。

下册按字母顺序给出了每条指令的详细信息,供编写汇编语言程序、调试程序、编译程序和操作系统时使用。

本书是 i486 程序员的必备参考资料。北京希望高级电脑技术公司资料部秦人华老师对本书的出版进行了周密的组织工作,希望本书能为读者提供较大帮助。

译者于北京
一九九一年八月

目 录

第一章 i486™ 处理器简介

1.1 本手册的编排	(2)
1.1.1 第一部分——应用程序设计	(2)
1.1.2 第二部分——系统程序设计	(2)
1.1.3 第三部分——数值处理	(3)
1.1.4 第四部分——兼容性	(3)
1.1.5 第五部分——指令集	(4)
1.1.6 附录部分	(4)
1.2 参考文献	(4)
1.3 有关约定	(5)
1.3.1 位和字节顺序	(5)
1.3.2 未定义的位和软件兼容性	(5)
1.3.3 指令的操作数	(6)
1.3.4 十六进制数	(6)
1.3.5 分段寻址	(6)
1.3.6 异 常	(6)

第一部分 应用程序设计

第二章 基本程序设计模式

2.1 内存组织	(8)
2.1.1 未分段或“平板”模式	(9)
2.1.2 分段模式	(9)
2.2 数据类型	(9)
2.3 寄存器	(11)
2.3.1 通用寄存器	(13)
2.3.2 段寄存器	(15)
2.3.3 栈的实现	(16)
2.4 指令格式	(19)
2.5 操作数的选择	(20)
2.5.1 立即操作数	(21)
2.5.2 寄存器操作数	(22)
2.5.3 内存操作数	(23)
2.6 中断和异常	(24)

第三章 应用程序设计

3.1 数据传送指令	(28)
3.1.1 通用数据传送指令	(28)

3.1.2	栈操作指令	(29)
3.1.3	类型转换指令	(29)
3.2	二进制算术指令	(31)
3.2.1	加法和减法指令	(32)
3.2.2	比较和符号转换指令	(33)
3.2.3	乘法指令	(33)
3.2.4	除法指令	(34)
3.3	十进制算术指令	(34)
3.3.1	压缩 BCD 的调节指令	(34)
3.3.2	非压缩 BCD 调节指令	(35)
3.4	逻辑指令	(35)
3.4.1	布尔运算指令	(35)
3.4.2	位测试和修改指令	(36)
3.4.3	位扫描指令	(36)
3.4.4	移位和循环移位指令	(36)
3.4.5	字节的条件设置指令	(44)
3.4.6	测试指令	(44)
3.5	转移控制指令	(45)
3.5.1	无条件转移指令	(46)
3.5.2	条件转移指令	(46)
3.5.3	软件中断	(47)
3.6	串运算	(48)
3.6.1	重复前缀	(49)
3.6.2	变址和方向标志控制	(49)
3.6.3	串指令	(49)
3.7	用于块结构语言的指令	(50)
3.8	标志控制指令	(54)
3.8.1	进位和方向标志控制指令	(55)
3.8.2	标志转移指令	(55)
3.9	数值指令	(56)
3.10	段寄存器指令	(56)
3.10.1	段寄存器转移指令	(57)
3.10.2	远转移控制指令	(58)
3.10.3	数据指针指令	(58)
3.11	其它指令	(58)
3.11.1	地址计算指令	(58)
3.11.2	空操作指令	(59)
3.11.3	变换指令	(59)
3.11.4	字节交换指令	(59)

3.11.5 交换及相加指令	(60)
3.11.6 比较并交换指令	(60)

第二部分 系统程序设计

第四章 系统体系结构

4.1 系统寄存器	(64)
4.1.1 系统标志	(64)
4.1.2 内存管理寄存器	(66)
4.1.3 控制寄存器	(66)
4.1.4 调试寄存器	(68)
4.2 系统指令	(69)

第五章 内存管理

5.1 选择分段模式	(73)
5.1.1 平板模式	(73)
5.1.2 受保护的平板模式	(74)
5.1.3 多段模式	(74)
5.2 段的转换	(75)
5.2.1 段寄存器	(75)
5.2.2 段选择子	(76)
5.2.3 段描述子	(78)
5.2.4 段描述子表	(82)
5.2.5 描述子表基址寄存器	(82)
5.3 页式转换	(83)
5.3.1 PG 位开放页机构	(84)
5.3.2 线性地址	(85)
5.3.3 页表	(85)
5.3.4 页表项	(85)
5.3.5 转换旁视缓冲器	(87)
5.4 组合的段页式转换	(87)
5.4.1 平板模式	(87)
5.4.2 跨越多页的段	(87)
5.4.3 跨越多段的页	(88)
5.4.4 对齐的段页边界	(88)
5.4.5 每段一个页表	(88)

第六章 保护

6.1 段一级的保护	(90)
6.2 段描述子和保护	(90)
6.2.1 类型检查	(91)
6.2.2 限长检查	(92)

6.2.3 特权级	(93)
6.3 限制数据的存取	(94)
6.3.1 存取代码段中的数据	(95)
6.4 限制控制的转移	(96)
6.5 门描述子	(97)
6.5.1 任务切换	(99)
6.5.2 过程的返回	(101)
6.6 操作系统保留的指令	(103)
6.6.1 特权指令	(103)
6.6.2 敏感指令	(104)
6.7 用于检查的指令	(104)
6.7.1 描述子的合法性	(105)
6.7.2 指针完整性和 RPL	(105)
6.8 页面保护	(106)
6.8.1 页表项包括保护参数	(106)
6.8.2 两级页表的组合保护	(107)
6.8.3 页保护的超越情况	(108)
6.9 段和页的组合保护	(108)
第七章 多任务	
7.1 任务状态段	(109)
7.2 TSS 描述子	(110)
7.3 任务寄存器	(110)
7.4 任务门描述子	(111)
7.5 任务切换	(112)
7.6 任务连接	(116)
7.6.1 忙位阻止循环	(117)
7.6.2 修改任务的连接状态	(118)
7.7 任务地址空间	(118)
7.7.1 任务线性物理空间	(118)
7.7.2 任务逻辑地址空间	(119)
第八章 输入输出	
8.1 I/O 寻址	(120)
8.1.1 I/O 地址空间	(120)
8.1.2 内存映象 I/O	(121)
8.2 I/O 指令	(122)
8.2.1 寄存器 I/O 指令	(122)
8.2.2 块 I/O 指令	(123)
8.3 保护与 I/O	(123)
8.3.1 I/O 特权级	(123)

8.3.2 I/O 权限位映象	110
-----------------------	-----

第九章 异常与中断

9.1 异常和中断向量	117
9.2 指令再启动	127
9.3 开放与关闭中断	128
9.3.1 NMI 屏蔽多重不可屏蔽中断	128
9.3.2 IF 屏蔽 INTR 中断	128
9.3.3 RF 屏蔽调试故障	128
9.3.4 对 SS 进行 MOV、POP 可屏蔽某些异常和中断	128
9.4 异常和中断的优先级	129
9.5 中断描述子表	129
9.6 IDT 描述子	130
9.7 中断门和中断处理	131
9.7.1 中断过程	131
9.7.2 中断任务	133
9.8 错误代码	134
9.9 异常条件	135
9.9.1 中断 0——除法错	135
9.9.2 调试异常	136
9.9.3 中断 3——断点	136
9.9.4 中断 4——溢出	136
9.9.5 中断 5——界限检查	137
9.9.6 中断 6——非法操作码	137
9.9.7 中断 7——设备不可用	137
9.9.8 中断 8——双故障	138
9.9.9 中断 9——Intel 保留	139
9.9.10 中断 10——非法 TSS	139
9.9.11 中断 11——段不存在	140
9.9.12 中断 12——栈异常	141
9.9.13 中断 13——通用保护	141
9.9.14 中断 14——页故障	142
9.9.15 中断 16——浮点错误	143
9.9.16 中断 17——对齐检查	143
9.10 异常小结	144
9.11 错误代码小结	145
第十章 初始化	
10.1 初始化后处理器的状态	147
10.2 实方式下软件初始化	148
10.2.1 系统表	149

10.2.2 NMI 中断	(149)
10.2.3 断点指令	(149)
10.2.4 开放高速缓存	(149)
10.3 切换至保护方式	(150)
10.3.1 系统表	(150)
10.3.2 NMI 中断	(150)
10.3.3 PE 位	(150)
10.4 保护方式下的软件初始化	(150)
10.4.1 分段	(150)
10.4.2 分页	(151)
10.4.3 任务	(151)
10.5 TLB 测试	(151)
10.5.1 TLB 结构	(151)
10.5.2 测试寄存器	(152)
10.5.3 测试操作	(153)
10.6 高速缓存 (Cache) 测试	(154)
10.6.1 Cache 结构	(154)
10.6.2 测试寄存器	(155)
10.7 初始化举例	(157)

第十一章 调试

11.1 调试支持	(163)
11.2 调试寄存器	(163)
11.2.1 调试地址寄存器 (DR0~DR3)	(164)
11.2.2 调试控制寄存器 (DR7)	(164)
11.2.3 调试状态寄存器 (DR6)	(165)
11.2.4 断点域识别	(165)
11.3 调试异常	(165)
11.3.1 中断 1——调试异常	(166)
11.3.2 中断 3——断点指令	(168)

第十二章 高速缓存 Cache

12.1 高速缓存介绍	(169)
12.2 内部 Cache 操作	(170)
12.2.1 Cache 禁止位	(170)
12.2.2 Cache 管理指令	(170)
12.2.3 自身修改代码	(170)
12.3 页级 Cache 管理	(171)
12.3.1 Cache 管理位	(171)

第十三章 多机处理

13.1 锁定和伪锁定的总线周期	(172)
------------------------	-------

13.1.1	LOCK 前缀和 LOCK # 信号	(172)
13.1.2	自动锁定	(173)
13.1.3	伪锁	(173)

第三部分 数值处理

第十四章 数值应用程序介绍

14.1	历史	(175)
14.2	性能	(175)
14.3	应用简便性	(176)
14.4	应用程序	(177)
14.5	程序设计接口	(178)

第十五章 浮点部件的结构

15.1	数值寄存器	(181)
15.1.1	FPU 寄存器栈	(181)
15.1.2	FPU 状态字	(182)
15.1.3	控制字	(184)
15.1.4	FPU 标记字	(185)
15.1.5	数值指令和数据指针	(186)
15.2	计算基础	(187)
15.2.1	数字系统	(187)
15.2.2	数据类型和格式	(188)
15.2.3	舍入控制	(192)
15.2.4	精度控制	(193)

第十六章 特殊的计算情况

16.1	特殊的数值	(194)
16.1.1	浮点实数	(194)
16.1.2	零	(198)
16.1.3	无穷数	(200)
16.1.4	NaN(“不是数”)	(202)
16.1.5	不定数	(203)
16.1.6	数据类型的编码形式	(204)
16.1.7	未支持的格式	(204)
16.2	数值异常	(208)
16.2.1	处理数值异常	(209)
16.2.2	非法运算	(210)
16.2.3	除零	(211)
16.2.4	非规格化操作数	(212)
16.2.5	算术上溢和下溢	(212)
16.2.6	非精确(精度)	(214)

16.2.7 异常的优先级.....	(215)
16.2.8 标准上溢/下溢异常处理程序	(215)
第十七章 浮点指令集	
17.1 目的操作数和源操作数.....	(216)
17.2 数据传送指令.....	(216)
17.3 基本算术指令.....	(217)
17.4 比较指令.....	(218)
17.5 超越数指令.....	(219)
17.6 常数指令.....	(220)
17.7 控制指令.....	(220)
第十八章 数值应用程序	
18.1 编程工具.....	(222)
18.1.1 高级语言.....	(222)
18.1.2 C 程序	(222)
18.1.3 PL/M-386/486	(222)
18.1.4 ASM386/486	(224)
18.1.5 示范程序.....	(227)
18.2 并发处理.....	(231)
18.2.1 并发的管理.....	(232)
第十九章 系统一级的考虑	
19.1 体系结构.....	(234)
19.1.1 独立的寻址方式.....	(234)
19.2 处理器的初始化和控制.....	(234)
19.2.1 系统的初始化.....	(234)
19.2.2 设置数值环境.....	(234)
19.2.3 初始化 FPU	(235)
19.2.4 仿真.....	(235)
19.2.5 处理数值异常.....	(236)
19.2.6 响应多重异常.....	(236)
19.2.7 异常恢复示例.....	(236)
第二十章 数值示范程序	
20.1 条件分支程序.....	(238)
20.2 异常处理程序.....	(238)
20.3 浮点数到 ASCII 码的转换程序.....	(242)
20.3.1 功能划分.....	(242)
20.3.2 对异常的考虑.....	(243)
20.3.3 特殊指令.....	(257)
20.3.4 操作的描述.....	(257)
20.3.5 数值的调节.....	(257)

20.3.6 输出格式.....	(58)
20.4 三角运算程序.....	(58)

第四部分 兼容性

第二十一章 运行 80286 和 386DX 或 SX CPU 程序

21.1 运行 80286CPU 任务的两种方式.....	(70)
21.2 与 80286CPU 的差别.....	(70)
21.2.1 24 位物理地址空间的环境.....	(70)
21.2.2 段描述子的保留字.....	(71)
21.2.3 新段描述子类型代码.....	(71)
21.2.4 限制 LOCK 前缀的语义.....	(71)
21.2.5 增加的异常.....	(71)
21.3 与 386™CPU 的差别.....	(71)
21.3.1 新的标志.....	(72)
21.3.2 新的异常.....	(72)
21.3.3 新的指令.....	(72)
21.3.4 新的控制寄存器位.....	(72)
21.3.5 新的页表项位.....	(72)
21.3.6 装入段描述子时的变化.....	(72)

第二十二章 实地址方式

22.1 地址的形式.....	(73)
22.2 寄存器和指令.....	(74)
22.3 中断和异常处理.....	(74)
22.4 进入和退出实地址方式.....	(75)
22.4.1 进入保护方式.....	(75)
22.5 返回实地址方式.....	(75)
22.6 实地址方式异常.....	(76)
22.7 与 8086CPU 的差别.....	(77)
22.8 与 80286CPU 实地址方式的差别.....	(80)
22.8.1 总线锁定.....	(80)
22.8.2 指令的存贮位置.....	(80)
22.8.3 通用寄存器的初始值.....	(80)
22.8.4 总线保持.....	(80)
22.8.5 与算逻协处理器的差别.....	(80)
22.9 与 386™DX CPU 实方式的差别.....	(81)
22.10 处理器类型检测代码.....	(81)

第二十三章 虚拟 8086 方式

23.1 运行 8086CPU 代码.....	(82)
23.1.1 寄存器和指令.....	(82)

23.1.2	地址转换	(283)
23.2	V-8086 任务的结构	(283)
23.2.1	V-8086 任务的分页	(284)
23.2.2	V-8086 任务的保护	(284)
23.3	进入和退出 V-8086 方式	(285)
23.3.1	通过任务切换的转移	(286)
23.3.2	通过陷阱门和中断门的转移	(286)
23.4	补充的敏感指令	(287)
23.4.1	仿真 8086 操作系统调用	(287)
23.4.2	仿真可中断标志位	(288)
23.5	虚拟 I/O	(288)
23.5.1	I/O 映象的 I/O	(288)
23.5.2	存储映象的 I/O	(288)
23.5.3	特殊 I/O 缓冲区	(289)
23.6	与 8086CPU 的区别	(289)
23.7	与实地址方式下的 80286CPU 的区别	(291)
23.7.1	特权级	(291)
23.7.2	总线封锁	(291)
23.8	与 386™DX 和 SX CPU 的区别	(292)
第十四章 混合 16 位和 32 位的代码		
24.1	使用 16 位和 32 位环境	(293)
24.2	混合 16 位和 32 位的操作	(294)
24.3	混合长度代码段中数据的共享	(294)
24.4	混合长度代码段中的控制转移	(295)
24.4.1	代码段指针的长度	(295)
24.4.2	控制转移的栈管理	(295)
24.4.3	中断控制转移	(297)
24.4.4	参量转换	(297)
24.4.5	接口过程	(297)
第二十五章 与 387、80287 和 8087 协处理器的兼容性		
25.1	与 386™CPU/387™NPX 系统的区别	(299)
25.2	与 80286/80287 系统的区别	(300)
25.2.1	数据类型与异常处理	(300)
25.2.2	特征字、状态字与控制字	(304)
25.2.3	指令设置	(306)
25.3	与 8086/8087 系统的区别	(309)

第五部分 i486 指令集

第二十六集 指令集

26.1 操作数的地址长度属性.....	(312)
26.1.1 缺省段属性.....	(312)
26.1.2 操作数长度和地址长度指令前缀.....	(312)
26.1.3 堆栈的地址长度属性.....	(313)
26.2 指令格式.....	(313)
26.2.1 ModR/M 和 SIB 字节.....	(314)
26.2.2 如何阅读指令集.....	(318)
AAA	(325)
ADD	(326)
AAM	(327)
AAS	(327)
ADC	(328)
ADD	(329)
AND	(330)
ARPL	(330)
BOUND	(331)
BSF	(332)
BSR	(333)
BSWAP	(334)
BT	(334)
BTC	(335)
BTR	(336)
PTS	(337)
CALL	(339)
CBW/CWDE	(342)
CLC	(343)
CLD	(343)
CLI	(344)
CLTS	(344)
CMC	(345)
CMP	(345)
CMPS/CMPSB/CMPSD	(346)
CMPXCHG	(348)
CWD/CDC	(349)
DAA	(349)
DAS	(350)
DEC	(351)
DIV	(351)
ENTER	(352)
F2XM1	(354)
FABS	(354)
FADD/FADDP/FLADD	(355)
FBLD	(356)
FBSTP	(356)
FCH3	(357)
FCLEX/FNCLEX	(357)
FCOM/FCOMP/FCMPP	(358)
FCOS	(359)
FDECSTP	(360)
FDIV/FDIVP/FIDIV	(361)
FDIVR/FDIVPR/FIDIVR	(362)
FFREE	(362)
FICOM/FICOMP	(363)
FILD	(364)
FINCSTP	(365)
FINIT/FNINIT	(365)
FIST/FISTP	(366)
FLD	(367)
FLDI/FLD2F/FLD2E	(368)
FLDPI/FLDG2/FLDLN2/FLDZ	(368)
FLDCW	(369)
FLDENV	(369)
FMUL/FMULP/FIMUL	(370)
FNOP	(371)
FPATAN	(371)

PREM	(3372)	LGDT/LIDT	(418)
PREMI	(373)	LGS/LSS/LDS/LES/LFS	(420)
PTAN	(374)	LLDT	(421)
PNDENT	(375)	LMSW	(422)
PTOR	(376)	LOCK	(423)
SAVE/NSAVE	(376)	LODS/LOSB/LODSW/LODSD	(424)
SCALE	(377)	LOOP/LOOPcond	(425)
SEN	(378)	LSL	(426)
SENCOS	(379)	LTR	(427)
FSQRT	(380)	MOV	(429)
FST/FSTP	(380)	MOV	(430)
FSTCW/FNSTCW	(381)	MOVS/MOVSIB/	(431)
FSTENV/FNSTENV	(382)	MOVSW/MOVSDB	(431)
FSTSW/FNSTSW	(383)	MOVSB	(433)
FSUB/FSUBP/FISUB	(384)	MOVZX	(433)
FSUBR/FSUBPR/FISUBR	(385)	MUL	(434)
FUCOM/FUCOMP/FUCOMPP	(385)	NEG	(435)
FWAIT	(387)	NOF	(435)
FXAM	(388)	NOT	(436)
FXCH	(388)	OR	(437)
FXTRACT	(389)	OUT	(438)
FYL2X	(390)	OUTS/OUTSB/	(439)
FYL2XP1	(391)	OUTSW/OUTSD	(439)
HLT	(392)	POP	(440)
IDIV	(392)	POPA/POPAD	(442)
IMUL	(393)	POPF/POPFD	(443)
IN	(394)	PUSH	(444)
INC	(396)	PUSHA/PUSHAD	(445)
INS/INSB/INSW/INSD	(396)	PUSHF/PUSHFD	(446)
INT/INTO	(398)	RCL/RCR/ROR	(447)
INVD	(400)	PEP/REPE/PEPZ/	(449)
INVLPG	(401)	REPNE/REPZ	(449)
IRET/IRETD	(403)	RET	(451)
Jcc	(408)	SAHF	(454)
JMP	(411)	SAL/SAR/SHL/SHR	(455)
LAHF	(415)	SBB	(457)
LAR	(415)	SCAS/SCASB/SCASW/SCASD	(458)
LEA	(416)	SETcc	(459)
LEAVE	(418)	SGDT/SIDT	(460)

SHLD	(461)	SUB	(468)
SHRD	(462)	TEST	(470)
SLDT	(463)	VERR/VERW	(470)
SMSW	(464)	WAIT	(471)
STC	(465)	WBINVD	(472)
STD	(465)	XADD	(473)
STI	(465)	XCHG	(474)
STOS/STOSB/STOSW/STOSD	(466)	XLAT/XLATB	(474)
STR	(467)	XOR	(475)

附录 A 操作码映象

A.1 缩写的关键	(477)
A.2 编址方式的代码	(477)
A.3 操作数类型的代码	(477)
A.4 寄存器代码	(478)

附录 B 标志交叉引用

B.1 编码关键	(483)
----------------	-------

附录 C 状态标志归纳

C.1 状态标志	(485)
C.2 编码关键	(485)

附录 D 条件代码

D.1 条件的定义	(487)
-----------------	-------

附录 E 指令格式与时序

附录 F 数字异常小结

附录 G 代码优化

G.1 寻址方式	(500)
G.2 预取单元	(501)
G.3 缓存和代码定位	(501)
G.4 NOP 指令	(502)
G.5 整数指令	(503)
G.6 条件代码	(504)
G.7 串指令	(505)
G.8 浮点指针指令	(505)
G.9 前缀操作码	(506)
G.10 时钟覆盖	(506)
G.11 杂项用法准则	(506)