(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 103729219 A
(43)申请公布日 2014.04.16

(21)申请号 201310726268.9

(22)申请日 2013.12.25

(71)申请人 合肥联宝信息技术有限公司
地址 230601 安徽省合肥市经济技术开发区翠微路6号海恒大厦4楼418号

(72)发明人 郑红文

(74)专利代理机构 北京大成律师事务所 11352
代理人 王卫东

(51)Int.Cl.
G06F 9/445(2006.01)

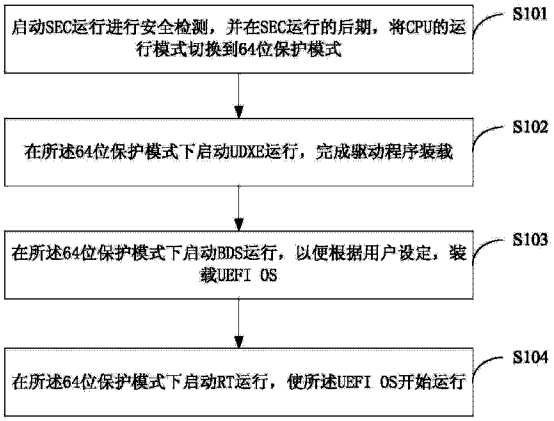
权利要求书2页 说明书4页 附图2页

(54)发明名称

一种UEFI BIOS架构方法及系统

(57)摘要

本发明公开了一种UEFI BIOS架构方法及系统，涉及计算机技术领域，其方法包括BIOS执行的以下步骤：启动SEC运行进行安全检测，并在SEC运行的后期，将CPU的运行模式切换到64位保护模式；然后，在所述64位保护模式下启动UDXE运行，完成驱动程序装载；接着，在所述64位保护模式下启动BDS运行，以便根据用户设定，装载UEFI OS；最后，在所述64位保护模式下启动RT运行，使所述UEFI OS开始运行；其中，所述BIOS是指基本输入/输出系统，所述UDXE是指统一可扩展固件接口。本发明UEFI BIOS代码全部运行在64位保护模式下，节省BIOS启动时间。



CN 103729219 A

1. 一种 UEFI BIOS 架构方法，其特征在于，包括 BIOS 执行的以下步骤：

启动 SEC 运行进行安全检测，并在 SEC 运行的后期，将 CPU 的运行模式切换到 64 位保护模式；

然后，在所述 64 位保护模式下启动 UDXE 运行，完成驱动程序装载；

接着，在所述 64 位保护模式下启动 BDS 运行，以便根据用户设定，装载 UEFI OS；

最后，在所述 64 位保护模式下启动 RT 运行，使所述 UEFI OS 开始运行；

其中，所述 BIOS 是指基本输入 / 输出系统，所述 UDXE 是指统一可扩展固件接口。

2. 根据权利要求 1 所述的方法，其特征在于，在所述 64 位保护模式下启动 UDXE 运行，完成驱动程序装载的步骤包括：

在所述 64 位保护模式下启动 UDXE 运行，并按照指令将所述 UDXE 内核装载到所述 CPU 的二级缓存中进行缓存；

所述缓存在所述 CPU 的二级缓存中的所述 UDXE 内核按照第一装载指令装载 64 位 UDXE 驱动程序分发器；

所述 64 位 UDXE 驱动程序分发器按照第二装载指令装载 64 位内存控制器驱动程序。

3. 根据权利要求 2 所述的方法，其特征在于，当所述装载 64 位内存控制器驱动程序后，将所述 UDXE 内核从所述 CPU 的二级缓存中移至 4M 内存中进行保存。

4. 根据权利要求 2 或 3 所述的方法，其特征在于，所述 64 位 UDXE 驱动程序分发器通过装载 BDS_ARCH_PROTOCOL 驱动程序，启动 BDS 运行。

5. 根据权利要求 4 所述的方法，其特征在于，还包括：

当所述 64 位 UDXE 驱动程序分发器装载所述 64 位内存控制器驱动程序或装载所述 BDS_ARCH_PROTOCOL 驱动程序失败时，结束运行工作。

6. 一种 UEFI BIOS 架构系统，其特征在于，包括 BIOS 执行的以下模块：

SEC 模块，用于启动 SEC 运行进行安全检测，并在 SEC 运行的后期，将 CPU 的运行模式切换到 64 位保护模式；

UDXE 模块，用于在所述 64 位保护模式下启动 UDXE 运行，完成驱动程序装载；

BDS 模块，用于在所述 64 位保护模式下启动 BDS 运行，以便根据用户设定，装载 UEFI OS；

RT 模块，用于在所述 64 位保护模式下启动 RT 运行，使所述 UEFI OS 开始运行；

其中，所述 BIOS 是指基本输入 / 输出系统，所述 UDXE 是指统一可扩展固件接口。

7. 根据权利要求 6 所述的系统，其特征在于，所述 UDXE 模块包括：

缓存单元，用于在所述 64 位保护模式下启动 UDXE 运行，并按照指令将所述 UDXE 内核装载到所述 CPU 的二级缓存中进行缓存；

装载单元，用于所述缓存在所述 CPU 的二级缓存中的所述 UDXE 内核按照第一装载指令装载 64 位 UDXE 驱动程序分发器；以及所述 64 位 UDXE 驱动程序分发器按照第二装载指令装载 64 位内存控制器驱动程序。

8. 根据权利要求 7 所述的系统，其特征在于，当所述装载 64 位内存控制器驱动程序后，将所述 UDXE 内核从所述 CPU 的二级缓存中移至 4M 内存中进行保存。

9. 根据权利要求 7 或 8 所述的系统，其特征在于，所述 64 位 UDXE 驱动程序分发器通过装载 BDS_ARCH_PROTOCOL 驱动程序，启动 BDS 运行。

10. 根据权利要求 9 所述的系统，其特征在于，还包括：

当所述 64 位 UDXE 驱动程序分发器装载所述 64 位内存控制器驱动程序或装载所述 BDS_ARCH_PROTOCOL 驱动程序失败时，结束运行工作。

一种 UEFI BIOS 架构方法及系统

技术领域

[0001] 本发明涉及计算机技术领域，特别涉及一种 UEFI(Unified Extensible Firmware Interface，统一的可扩展固件接口) BIOS (Basic Input Output System，基本输入输出系统) 架构的方法及系统。

背景技术

[0002] 目前的 UEFI BIOS 架构由五部份构成 :SEC (Security phase，安全阶段)，PEI(Pre-EFI Initialization phase，EFI 初始化准备阶段)，DXE (Driver Execution Environment phase，驱动执行环境阶段)，BDS (Boot Device Sele ction phase，启动设备选择阶段) 和 RT (Run Time phase，运行时期阶段)。SEC 运行在大实模式，而 PEI 运行在 32 位保护模式下，然后通过 DXE IPL (Ini tial Program Load，初始程序导入) 模块将 CPU 切换到 64 位保护模式中的长模式。随后的 DXE、BDS 和 RT 均运行在 64 位保护模式下。在 UEFI BIOS 架构中，我们需要增加一个模块 DXE IPL 来完成 CPU 模式从 32 位移到 64 位模式下。带来的问题是 :

[0003] 1.UEFI BIOS 代码一部分运行在 32 位保护模式下，而另一部份运行在 64 位保护模式下。这两部份代码必须分别编译，无法共享运行库。

[0004] 2.UEFI BIOS 要利用 DXE IPL 模块完成 32 位到 64 位保护模式的切换，这个切换过程会占用一部份 BIOS 启动时间。

[0005] 为解决上述问题，本发明提供了一种 UEFI BIOS 架构的方法及系统。

发明内容

[0006] 本发明的目的在于提供一种 UEFI BIOS 架构的方法及系统，解决了现有技术中需要进行 32 位到 64 位保护模式的切换，使得 BIOS 启动时间慢的问题。

[0007] 根据本发明的一个方面，提供了一种 UEFI BIOS 架构的方法，包括 BIOS 执行的以下步骤 :

[0008] 启动 SEC 运行进行安全检测，并在 SEC 运行的后期，将 CPU 的运行模式切换到 64 位保护模式 ;

[0009] 然后，在所述 64 位保护模式下启动 UDXE (Unified Driver Execution Environment，统一的驱动执行环境) 运行，完成驱动程序装载 ;

[0010] 接着，在所述 64 位保护模式下启动 BDS 运行，以便根据用户设定，装载 UEFI OS (Operation System，操作系统) ;

[0011] 最后，在所述 64 位保护模式下启动 RT 运行，使所述 UEFI OS 开始运行 ;

[0012] 其中，所述 BIOS 是指基本输入 / 输出系统，所述 UDXE 是指统一可扩展固件接口。

[0013] 优选地，在所述 64 位保护模式下启动 UDXE 运行，完成驱动程序装载的步骤包括 :

[0014] 在所述 64 位保护模式下启动 UDXE 运行，并按照指令将所述 UDXE 内核装载到所述 CPU 的二级缓存中进行缓存 ;

[0015] 所述缓存在所述 CPU 的二级缓存中的所述 UDXE 内核按照第一装载指令装载 64 位 UDXE 驱动程序分发器；

[0016] 所述 64 位 UDXE 驱动程序分发器按照第二装载指令装载 64 位内存控制器驱动程序。

[0017] 优选地，当所述装载 64 位内存控制器驱动程序后，将所述 UDXE 内核从所述 CPU 的二级缓存中移至 4M 内存中进行保存。

[0018] 优选地，所述 64 位 UDXE 驱动程序分发器通过装载 BDS_ARCH_PROTOCOL 驱动程序，启动 BDS 运行。

[0019] 优选地，还包括：

[0020] 当所述 64 位 UDXE 驱动程序分发器装载所述 64 位内存控制器驱动程序或装载所述 BDS_ARCH_PROTOCOL 驱动程序失败时，结束运行工作。

[0021] 根据本发明的另一方面，提供了一种 UEFI BIOS 架构的系统，包括 BIOS 执行的以下模块：

[0022] SEC 模块，用于启动 SEC 运行进行安全检测，并在 SEC 运行的后期，将 CPU 的运行模式切换到 64 位保护模式；

[0023] UDXE 模块，用于在所述 64 位保护模式下启动 UDXE 运行，完成驱动程序装 载；

[0024] BDS 模块，用于在所述 64 位保护模式下启动 BDS 运行，以便根据用户设定，装载 UEFI OS；

[0025] RT 模块，用于在所述 64 位保护模式下启动 RT 运行，使所述 UEFI OS 开始运行；

[0026] 其中，所述 BIOS 是指基本输入 / 输出系统，所述 UDXE 是指统一可扩展固件接口。

[0027] 优选地，所述 UDXE 模块包括：

[0028] 缓存单元，用于在所述 64 位保护模式下启动 UDXE 运行，并按照指令将所述 UDXE 内核装载到所述 CPU 的二级缓存中进行缓存；

[0029] 装载单元，用于所述缓存在所述 CPU 的二级缓存中的所述 UDXE 内核按照第一装载指令装载 64 位 UDXE 驱动程序分发器；以及所述 64 位 UDXE 驱动程序分发器按照第二装载指令装载 64 位内存控制器驱动程序。

[0030] 优选地，当所述装载 64 位内存控制器驱动程序后，将所述 UDXE 内核从所述 CPU 的二级缓存中移至 4M 内存中进行保存。

[0031] 优选地，所述 64 位 UDXE 驱动程序分发器通过装载 BDS_ARCH_PROTOCOL 驱动程序，启动 BDS 运行。

[0032] 优选地，还包括：

[0033] 当所述 64 位 UDXE 驱动程序分发器装载所述 64 位内存控制器驱动程序或装载所述 BDS_ARCH_PROTOCOL 驱动程序失败时，结束运行工作。

[0034] 与现有技术相比较，本发明的有益效果在于：

[0035] 本发明通过使用 UDXE，使得 UEFI BIOS 代码全部运行在 64 位保护模式下，节省了 BIOS 启动时间，提高了用户体验。

附图说明

[0036] 图 1 是本发明实施例提供的一种 UEFI BIOS 架构的方法流程图；

[0037] 图 2 是本发明实施例提供的一种 UEFI BIOS 架构的系统的示意图；

[0038] 图 3 是本发明实施例提供的 UEFI BIOS 的架构流程图。

具体实施方式

[0039] 以下结合附图对本发明的优选实施例进行详细说明，应当理解，以下所说明的优选实施例仅用于说明和解释本发明，并不用于限定本发明。

[0040] 图 1 显示了本发明实施例提供的一种 UEFI BIOS 架构的方法流程图，如图 1 所示，包括 BIOS 执行的以下步骤：

[0041] 步骤 S101：启动 SEC 运行进行安全检测，并在 SEC 运行的后期，将 CPU 的运行模式切换到 64 位保护模式；

[0042] 步骤 S102：在所述 64 位保护模式下启动 UDXE 运行，完成驱动程序装载；

[0043] 步骤 S103：在所述 64 位保护模式下启动 BDS 运行，以便根据用户设定，装载 UEFI OS；

[0044] 步骤 S104：在所述 64 位保护模式下启动 RT 运行，使所述 UEFI OS 开始运行；

[0045] 其中，所述 BIOS 是指基本输入 / 输出系统，所述 UDXE 是指统一可扩展固件接口。

[0046] 本发明在启动 SEC 运行进行安全检测，并在 SEC 运行的后期，将 CPU 的运行模式从大实模式切换到 64 位保护模式。

[0047] 本发明在所述 64 位保护模式下启动 UDXE 运行，完成驱动程序装载的步骤包括：在所述 64 位保护模式下启动 UDXE 运行，并按照指令将所述 UDXE 内核装载到所述 CPU 的二级缓存中进行缓存；所述缓存在所述 CPU 的二级缓存中的所述 UDXE 内核按照第一装载指令装载 64 位 UDXE 驱动程序分发器；所述 64 位 UDXE 驱动程序分发器按照第二装载指令装载 64 位内存控制器驱动程序。

[0048] 其中，当所述装载 64 位内存控制器驱动程序后，将所述 UDXE 内核从所述 CPU 的二级缓存中移至 4M 内存中进行保存。

[0049] 本发明所述 64 位 UDXE 驱动程序分发器通过装载 BDS_ARCH_PROTOCOL 驱动程序，启动 BDS 运行。

[0050] 本发明还包括：当所述 64 位 UDXE 驱动程序分发器装载所述 64 位内存控制器驱动程序或装载所述 BDS_ARCH_PROTOCOL 驱动程序失败时，结束运行工作。

[0051] 图 2 显示了本发明实施例提供的一种 UEFI BIOS 架构的系统的示意图，如图 2 所示，包括 BIOS 执行的以下模块：SEC 模块 201，用于启动 SEC 运行进行安全 检测，并在 SEC 运行的后期，将 CPU 的运行模式切换到 64 位保护模式；UDXE 模块 202，用于在所述 64 位保护模式下启动 UDXE 运行，完成驱动程序装载；BDS 模块 203，用于在所述 64 位保护模式下启动 BDS 运行，以便根据用户设定，装载 UEFI OS；RT 模块 204，用于在所述 64 位保护模式下启动 RT 运行，使所述 UEFIOS 开始运行；其中，所述 BIOS 是指基本输入 / 输出系统，所述 UDXE 是指统一可扩展固件接口。

[0052] 其中，所述 UDXE 模块 202 包括：缓存单元，用于在所述 64 位保护模式下启动 UDXE 运行，并按照指令将所述 UDXE 内核装载到所述 CPU 的二级缓存中进行缓存；装载单元，用于所述缓存在所述 CPU 的二级缓存中的所述 UDXE 内核按照第一装载指令装载 64 位 UDXE 驱动程序分发器；以及所述 64 位 UDXE 驱动程序分发器按照第二装载指令装载 64 位内存控制

器驱动程序。

[0053] 其中，当所述装载64位内存控制器驱动程序后，将所述UDXE内核从所述CPU的二级缓存中移至4M内存中进行保存。

[0054] 所述64位UDXE驱动程序分发器通过装载BDS_ARCH_PROTOCOL驱动程序，启动BDS运行。

[0055] 本发明还包括：当所述64位UDXE驱动程序分发器装载所述64位内存控制器驱动程序或装载所述BDS_ARCH_PROTOCOL驱动程序失败时，结束运行工作。

[0056] 图3显示了本发明实施例提供的UEFI BIOS的架构流程图，如图3所示，包括以下步骤：

[0057] 步骤1、BIOS模块启动进入SEC阶段后，在SEC最后阶段切换CPU模式为64位保护模式长模式。

[0058] 步骤2、BIOS将UDXE内核装载到CPU二级缓存，所述UDXE内核再装载64位UDXE驱动程序分发器，若加载成功，进入步骤3，若加载失败，结束运行工作。

[0059] 64位UDXE驱动程序分发器装载64位内存控制器驱动程序，完成内存初始化后，将所述UDXE内核从CPU二级缓存移到4M常规内存中，在所述常规内存中64位UDXE驱动程序分发器继续装载其它64位驱动程序，如显卡，南桥等。

[0060] 步骤3、64位UDXE驱动程序分发器装载BDS_ARCH_PROTOCOL驱动程序，进入BDS阶段，若加载成功，进入步骤4，若加载失败，结束运行工作。

[0061] 步骤4、BDS阶段根据用户设定，装载并运行UEFI OS从所选定启动分区上，若加载成功，进入步骤5，若加载失败，结束运行工作。

[0062] 步骤5、UEFI OS开始运行，进入RT阶段。

[0063] 综上所述，本发明的新型UEFI BIOS架构由四部份构成：SEC、UDXE、BDS及RT。SEC运行在大实模式，而UDXE运行在64位保护模式中的长模式，是新型UEFI BIOS架构的第二阶段，此阶段驱动程序全部为64位微软PE32+格式驱动程序，并且提供一个64位的驱动程序装载器，在随后的BDS及RT均运行在64位保护模式下。

[0064] 综上所述，本发明具有以下技术效果：

[0065] a.UEFI BIOS代码全部运行在64位保护模式下，不再需要32位模块。

[0066] b.无需再利用DXE IPL模块进行32位到64位保护模式切换，节省BIOS启动时间。

[0067] 尽管上文对本发明进行了详细说明，但是本发明不限于此，本技术领域技术人员可以根据本发明的原理进行各种修改。因此，凡按照本发明原理所作的修改，都应当理解为落入本发明的保护范围。

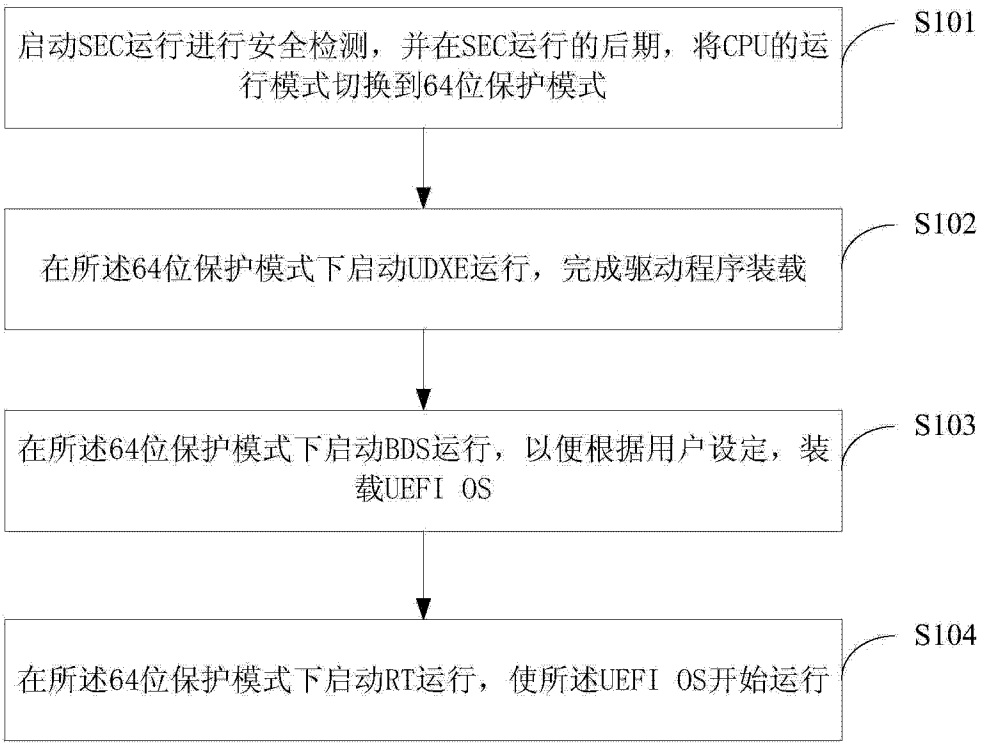

启动SEC运行进行安全检测，并在SEC运行的后期，将CPU的运行模式切换到64位保护模式
S101
在所述64位保护模式下启动UDXE运行，完成驱动程序装载
S102
在所述64位保护模式下启动BDS运行，以便根据用户设定，装载UEFI OS
S103
在所述64位保护模式下启动RT运行，使所述UEFI OS开始运行
S104

图 1

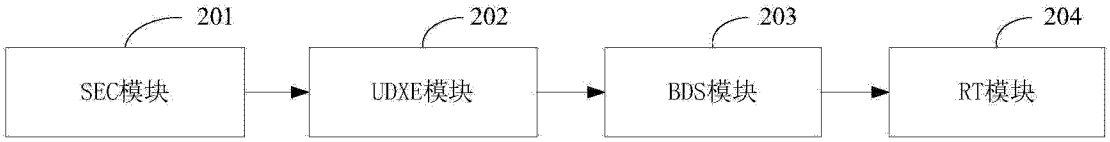

201
SEC模块
202
UDXE模块
203
BDS模块
204
RT模块

图 2

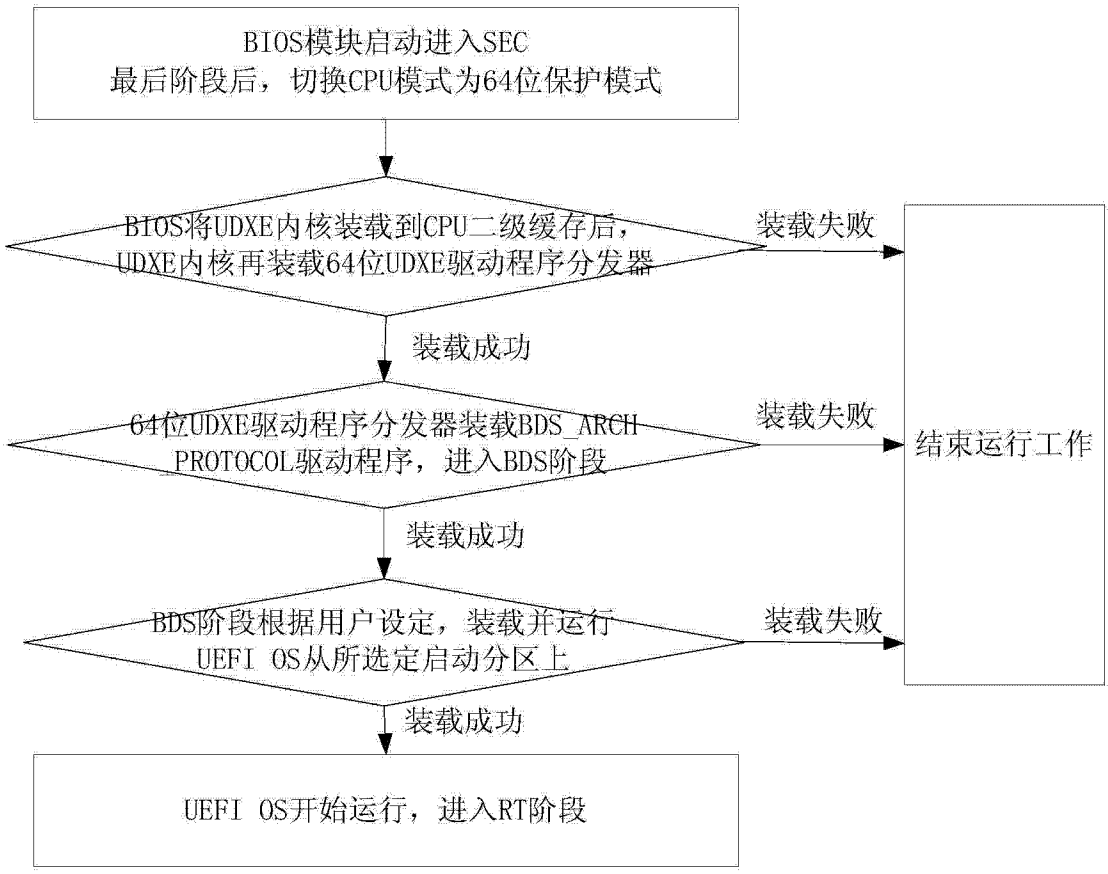

BIOS模块启动进入SEC
最后阶段后，切换CPU模式为64位保护模式
BIOS将UDXE内核装载到CPU二级缓存后，
UDXE内核再装载64位UDXE驱动程序分发器
装载失败
装载成功
64位UDXE驱动程序分发器装载BDS_ARCH
PROTOCOL驱动程序，进入BDS阶段
装载失败
装载成功
BDS阶段根据用户设定，装载并运行
UEFI OS从所选定启动分区上
装载失败
装载成功
UEFI OS开始运行，进入RT阶段
结束运行工作

图 3