



(12)发明专利申请

(10)申请公布号 CN 110888693 A

(43)申请公布日 2020.03.17

(21)申请号 201911157582.3

(22)申请日 2019.11.22

(71)申请人 烽火通信科技股份有限公司
地址 430000 湖北省武汉市东湖高新技术
开发区高新四路6号

(72)发明人 石王刚 姜超 叶剑

(74)专利代理机构 武汉智权专利代理事务所
(特殊普通合伙) 42225

代理人 陈文净

(51) Int. Cl.

G06F 9/448(2018.01)

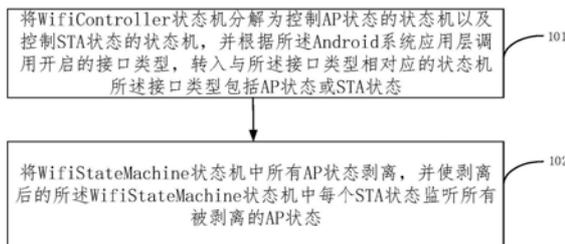
权利要求书2页 说明书7页 附图4页

(54)发明名称

一种在Android系统中支持AP和STA共存的方法及装置

(57)摘要

本发明实施例提供了一种在Android系统中支持AP和STA共存的方法及装置,包括:将WifiController状态机分解为控制AP状态的状态机以及控制STA状态的状态机,并根据所述Android系统应用层调用开启的接口类型,转入与所述接口类型相对应的状态机,所述接口类型包括AP状态或STA状态;将WifiStateMachine状态机中所有AP状态剥离,并使剥离后的所述WifiStateMachine状态机中每个STA状态监听所有被剥离的AP状态。本发明实施例通过将Wi-Fi状态机分解成两部分,分别独立连接处理应用程序和驱动层的指令,每一部分都能保障AP和STA的独立运行,避免由于Android框架层中的Wi-Fi状态机的错乱。



1. 一种在Android系统中支持AP和STA共存的方法,其特征在于,包括:

将WifiController状态机分解为控制AP状态的状态机以及控制STA状态的状态机,并根据所述Android系统应用层调用开启的接口类型,转入与所述接口类型相对应的状态机,所述接口类型包括AP状态或STA状态;

将WifiStateMachine状态机中所有AP状态剥离,并使剥离后的所述WifiStateMachine状态机中每个STA状态监听所有被剥离的AP状态。

2. 根据权利要求1所述的在Android系统中支持AP和STA共存的方法,其特征在于,所述将WifiStateMachine状态机中所有AP状态剥离,并使剥离后的所述WifiStateMachine状态机中每个STA状态监听所有被剥离的AP状态,包括:

将所述WifiStateMachine状态机中所有AP状态剥离并存储在内存中,并将内存地址分配给剥离后的WifiStateMachine状态机中每个STA状态,以使剥离后的WifiStateMachine状态机中每个STA状态监听所述内存。

3. 根据权利要求2所述的在Android系统中支持AP和STA共存的方法,其特征在于,所述方法还包括:

若所述接口状态为AP状态,则将当前AP状态值保存在所述内存中。

4. 根据权利要求2所述的在Android系统中支持AP和STA共存的方法,其特征在于,所述方法还包括:

若所述接口类型为STA状态,则直接转入剥离AP状态后的WifiStateMachine状态机。

5. 根据权利要求1所述的在Android系统中支持AP和STA共存的方法,其特征在于,在所述转入与所述接口类型相对应的状态机之后,所述方法还包括:

保持与所述接口类型不一致的状态机处于开启状态。

6. 一种在Android系统中支持AP和STA共存的装置,其特征在于,包括:

应用层对接模块,用于将WifiController状态机分解为控制AP状态的状态机以及控制STA状态的状态机,并根据所述Android系统应用层调用开启的接口类型,转入与所述接口类型相对应的状态机,所述接口类型包括AP状态或STA状态;

驱动层对接模块,用于将WifiStateMachine状态机中所有AP状态剥离,并使剥离后的所述WifiStateMachine状态机中每个STA状态监听所有被剥离的AP状态。

7. 根据权利要求6所述的在Android系统中支持AP和STA共存的装置,其特征在于,所述驱动层对接模块具体用于:

将所述WifiStateMachine状态机中所有AP状态剥离并存储在内存中,并将内存地址分配给剥离后的WifiStateMachine状态机中每个STA状态,以使剥离后的WifiStateMachine状态机中每个STA状态监听所述内存。

8. 根据权利要求7所述的在Android系统中支持AP和STA共存的装置,其特征在于,所述装置还包括:

AP状态存储模块,用于若所述接口状态为AP状态,则将当前AP状态值保存在所述内存中。

9. 根据权利要求7所述的在Android系统中支持AP和STA共存的装置,其特征在于,所述装置还包括:

STA状态处理模块,用于若所述接口类型为STA状态,则直接转入剥离AP状态后的

WifiStateMachine状态机。

10. 根据权利要求6所述的在Android系统中支持AP和STA共存的装置,其特征在于,所述装置还用于:

保持与所述接口类型不一致的状态机处于开启状态。

一种在Android系统中支持AP和STA共存的方法及装置

技术领域

[0001] 本发明涉及Android技术领域,尤其涉及一种在Android系统中支持AP和STA共存的方法及装置。

背景技术

[0002] 在原生Android系统中,无线网络,简称Wi-Fi是无法支持无线访问节点(Access Point,AP)与站点(Station,STA)共存的。

[0003] 具体的,在Android系统中,当应用程序打开了STA时,会调用Android框架接口将AP关闭,此时只能实现STA功能。而当应用程序打开了AP时,也会调用Android框架接口将STA关闭,此时只能实现AP功能。如果应用程序强制性的将STA和AP同时打开,会导致Android框架层中的Wi-Fi状态机错乱,将无法正常使用Wi-Fi功能。

[0004] 因此,现在亟需一种能够在Android系统中支持AP和STA共存的方法来解决上述问题。

发明内容

[0005] 为了解决上述问题,本发明实施例提供一种克服上述问题或者至少部分地解决上述问题的一种在Android系统中支持AP和STA共存的方法及装置。

[0006] 第一方面本发明实施例提供一种在Android系统中支持AP和STA共存的方法,包括:

[0007] 将WifiController状态机分解为控制AP状态的状态机以及控制STA状态的状态机,并根据所述Android系统应用层调用开启的接口类型,转入与所述接口类型相对应的状态机,所述接口类型包括AP状态或STA状态;

[0008] 将WifiStateMachine状态机中所有AP状态剥离,并使剥离后的所述WifiStateMachine状态机中每个STA状态监听所有被剥离的AP状态。

[0009] 其中,所述将WifiStateMachine状态机中所有AP状态剥离,并使剥离后的所述WifiStateMachine状态机中每个STA状态监听所有被剥离的AP状态,包括:

[0010] 将所述WifiStateMachine状态机中所有AP状态剥离并存储在内存中,并将内存地址分配给剥离后的WifiStateMachine状态机中每个STA状态,以使剥离后的WifiStateMachine状态机中每个STA状态监听所述内存。

[0011] 其中,所述方法还包括:

[0012] 若所述接口状态为AP状态,则将当前AP状态值保存在所述内存中。

[0013] 其中,所述方法还包括:

[0014] 若所述接口类型为STA状态,则直接转入剥离AP状态后的WifiStateMachine状态机。

[0015] 其中,在所述转入与所述接口类型相对应的状态机之后,所述方法还包括:

[0016] 保持与所述接口类型不一致的状态机处于开启状态。

[0017] 第二方面本发明实施例还提供一种在Android系统中支持AP和STA共存的装置,包括:

[0018] 应用层对接模块,用于将WifiController状态机分解为控制AP状态的状态机以及控制STA状态的状态机,并根据所述Android系统应用层调用开启的接口类型,转入与所述接口类型相对应的状态机,所述接口类型包括AP状态或STA状态;

[0019] 驱动层对接模块,用于将WifiStateMachine状态机中所有AP状态剥离,并使剥离后的所述WifiStateMachine状态机中每个STA状态监听所有被剥离的AP状态。

[0020] 其中,所述驱动层对接模块具体用于:

[0021] 将所述WifiStateMachine状态机中所有AP状态剥离并存储在内存中,并将内存地址分配给剥离后的WifiStateMachine状态机中每个STA状态,以使剥离后的WifiStateMachine状态机中每个STA状态监听所述内存。

[0022] 其中,所述装置还包括:

[0023] AP状态存储模块,用于若所述接口状态为AP状态,则将当前AP状态值保存在所述内存中。

[0024] 其中,所述装置还包括:

[0025] STA状态处理模块,用于若所述接口类型为STA状态,则直接转入剥离AP状态后的WifiStateMachine状态机。

[0026] 其中,所述装置还用于:

[0027] 保持与所述接口类型不一致的状态机处于开启状态。

[0028] 第三方面本发明实施例提供了一种电子设备,包括:

[0029] 处理器、存储器、通信接口和总线;其中,所述处理器、存储器、通信接口通过所述总线完成相互间的通信;所述存储器存储有可被所述处理器执行的程序指令,所述处理器调用所述程序指令能够执行上述在Android系统中支持AP和STA共存的方法。

[0030] 第四方面本发明实施例提供了一种非暂态计算机可读存储介质,所述非暂态计算机可读存储介质存储计算机指令,所述计算机指令使所述计算机执行上述在Android系统中支持AP和STA共存的方法。

[0031] 本发明实施例提供的一种在Android系统中支持AP和STA共存的方法及装置,通过将Android系统框架层中的Wi-Fi状态机分解成两部分,分别独立连接处理应用程序和驱动层的指令,每一部分都能保障AP和STA的独立运行,避免由于Android框架层中的Wi-Fi状态机的错乱。

附图说明

[0032] 为了更清楚地说明本发明实施例或现有技术中的技术方案,下面将对实施例或现有技术描述中所需要使用的附图作一简单地介绍,显而易见地,下面描述中的附图是本发明的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其他的附图。

[0033] 图1是本发明实施例提供的一种在Android系统中支持AP和STA共存的方法流程示意图;

[0034] 图2是原生Android系统WifiController状态机包括的状态图;

- [0035] 图3是本发明实施例提供的WifiController分解后的状态机示意图；
- [0036] 图4是原生态Android系统中WifiStateMachine状态机包括的状态图；
- [0037] 图5是本发明实施例提供的WifiStateMachine剥离状态示意图；
- [0038] 图6是本发明实施例提供的一种在Android系统中支持AP和STA共存的装置结构示意图；
- [0039] 图7是本发明实施例提供的一种电子设备的结构框图。

具体实施方式

[0040] 为使本发明实施例的目的、技术方案和优点更加清楚，下面将结合本发明实施例中的附图，对本发明实施例中的技术方案进行清楚地描述，显然，所描述的实施例是本发明一部分实施例，而不是全部的实施例。基于本发明中的实施例，本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例，都属于本发明保护的范围。

[0041] 图1是本发明实施例提供的一种在Android系统中支持AP和STA共存的方法流程图示意图，如图1所示，包括：

[0042] 101、将WifiController状态机分解为控制AP状态的状态机以及控制STA状态的状态机，并根据所述Android系统应用层调用开启的接口类型，转入与所述接口类型相对应的状态机，所述接口类型包括AP状态或STA状态；

[0043] 102、将WifiStateMachine状态机中所有AP状态剥离，并使剥离后的所述WifiStateMachine状态机中每个STA状态监听所有被剥离的AP状态。

[0044] 由背景的内容可知，在原生Android系统中，使用Wi-Fi功能时是不能够支持AP和STA共存的，也就是说当你使用的应用程序是需要调用AP接口时，需要将STA接口关闭，反之使用的应用程序需要调用STA接口时，也需要将AP接口关闭，因为如果强制性保持AP和STA接口同时打开，会使框架层中Wi-Fi状态机状态辨别不清，从而状态错乱，无法正常开启Wi-Fi功能。

[0045] 针对上述背景，本发明实施例提供一种基于Android系统下Wi-Fi中同时支持AP和STA共存的方式，即应用程序可以同时使用AP功能和STA功能，同时可以正常进行Wi-Fi上网。需要说明的是，Android系统可以分为多个层级，由底层至上可以包括有驱动层、框架Framework层、应用Application层，具体在框架层是为开发人员将一些基本功能实现，通过接口提供给上层调用，可以重复的调用的具体实现层，那么在框架层中一般会定义Wi-Fi状态机，可以理解的是，状态机就是一个定义了很多状态的机器，它收到消息后，会根据消息来切换这个机器的状态，状态类型包括两种，AP类型和STA类型。本发明实施例对接Android系统应用层的状态机采用WifiController状态机，对接Android系统驱动层的状态机采用WifiStateMachine状态机。WifiController状态机是高级别的Wi-Fi状态机，它管理的状态是Wi-Fi开关、Wi-Fi热点开关等状态，更适用于与应用层对接，WifiStateMachine状态机则表述wifi更加细致的状态，更适用于与驱动层对接。

[0046] 具体的，在步骤101中，图2是原生Android系统WifiController状态机包括的状态图，如图2所示，原生Android系统WifiController状态机包括有大量的状态示例，每一个状态底侧代表着该状态所属为AP状态或STA状态，其中，DefaultState状态既包括AP状态也包括STA状态。对于该原生态WifiController状态机，本发明实施例将其分解为只控制AP状态

的AP状态机,在本发明实施例中称为WifiAPController状态机,以及只控制STA状态的STA状态机,在本发明实施例中称为WifiSTAController状态机。图3是本发明实施例提供的WifiController分解后的状态机示意图,如图3所示,虚线上方为本发明实施例分离出的WifiSTAController状态机,虚线下方为本发明实施例分离出的WifiAPController状态机。那么当有来自于应用层操作消息时,通过识别消息操作的接口类型即可选择相应的状态机完成操作,例如:应用层调用开启的是AP状态接口,则相应的转入WifiAPController状态机,相反的,如果应用层调用开启的是STA状态接口,则相应的接入WifiSTAController状态机。从上述过程可以看出,本发明实施例通过将对接应用层的WifiController状态机分解为只处理AP状态的WifiAPController状态机与只处理STA状态的WifiSTAController状态机,实现了AP和STA的独立运行,尽量减少了相互之间的干扰。

[0047] 最后,在步骤102中,本发明实施例除了对WifiController状态机处理,还对WifiStateMachine状态机进行了处理,图4是原生态Android系统中WifiStateMachine状态机包括的状态图,如图4所示,同样的在WifiStateMachine状态机中包括的各个状态也属于AP状态或STA状态,也有DefaultState状态既包括AP状态也包括STA状态。那么对于此种情况,本发明实施例会将AP状态进行剥离,由于WifiStateMachine状态机中STA的状态数量要远多于AP状态数量,那么为了系统实现方便以及最小化运算量,本发明实施例会将所有AP状态进行剥离,而保持原WifiStateMachine状态机中STA状态机不变。图5是本发明实施例提供的WifiStateMachine剥离状态示意图,如图5所示,虚线上方是本发明实施例剥离出的所有包含AP的状态,在本发明实施例中将这些状态统称为APState,然后剥离后的WifiStateMachine中则没有了上述状态,剩下的都只是属于STA的状态,然后为每个属于的STA状态均设置一个APState的监听功能,从而能够在使用任意一个STA状态时都能同时对AP功能进行处理,实现了STA状态和AP状态的共存。以处理AP状态为例,当应用层处理AP状态时,驱动层会使用AP状态机记录当前AP状态值,然后为了能够此时还能使用STA状态机,本发明实施例会把当前记录的AP状态值提出来存储为APState,然后APState的存储路径分配给剥离后WifiStateMachine中剩下的每个STA状态,从而剥离后WifiStateMachine中剩下的每个STA状态都能对AP状态进行监听,当此时需要使用STA状态时,直接从STA状态机中就能读出AP状态,从而能够同时使用STA和AP。

[0048] 本发明实施例提供的一种在Android系统中支持AP和STA共存的方法,通过将Android系统框架层中的Wi-Fi状态机分解成两部分,分别独立连接处理应用程序和驱动层的指令,每一部分都能保障AP和STA的独立运行,避免由于Android框架层中的Wi-Fi状态机的错乱。

[0049] 在上述实施例的基础上,将所述WifiStateMachine状态机中所有AP状态剥离,并使剥离后的所述WifiStateMachine状态机中每个STA状态监听所有被剥离的AP状态,包括:

[0050] 将所述WifiStateMachine状态机中所有AP状态剥离并存储在内存中,并将内存地址分配给剥离后的WifiStateMachine状态机中每个STA状态,以使剥离后的WifiStateMachine状态机中每个STA状态监听所述内存。

[0051] 由上述实施例的内容可知,本发明实施例处理WifiStateMachine状态机时是将状态剥离后再进行监听。那么剥离的状态本发明实施例优选的会将其存储在内存中,可以理解的是,在内存中能够方便的保存AP状态值或STA状态值,并且对剥离后的

WifiStateMachine状态机中每一个状态分配监听对象时,能够方便的直接将内存地址进行分配,从而可以直接由内存中读取当前保存的AP状态值或STA状态值,方便且快捷的使得AP状态和STA状态共存。

[0052] 在上述实施例的基础上,所述方法还包括:

[0053] 若所述接口状态为AP状态,则将当前AP状态值保存在所述内存中。

[0054] 由上述实施例的内容可知,本发明实施例会在驱动层进行状态剥离,且剥离出的状态为AP状态。那么当应用层有AP状态接入进行处理时,在应用层会直接接入WifiAPController状态机,而在驱动层会将WifiStateMachine状态机中当前所有AP状态剥离,并将当前AP状态值进行存储,然后将存储路径分配给每个剥离后WifiStateMachine状态机中每个STA状态,从而每个STA状态都能对每个AP状态进行状态值获取,从而实现每个STA状态都能监听AP状态。那么在操作任何一类STA状态时,都能在该STA状态中去捕获AP的当前状态,从而实现了对AP状态和STA状态的同时处理。

[0055] 在上述实施例的基础上,所述方法还包括:

[0056] 若所述接口类型为STA状态,则直接转入剥离AP状态后的WifiStateMachine状态机。

[0057] 由上述实施例的内容可知,本发明实施例需要解决的问题是实现AP和STA状态的共存,对于应用层而言,接入的类型可能为AP状态,也可能为STA状态,当接入类型为STA状态时,应用层会直接转入WifiSTAController状态机,而在驱动层中WifiStateMachine状态机只被剥离了AP状态,剩下的STA状态机仍能照常处理,那么可以直接转入剥离AP状态后的WifiStateMachine状态机对STA状态进行处理即可,与此同时,在对STA状态处理时,由于剥离AP状态后的WifiStateMachine状态机中每个STA状态都会对AP状态进行监听,那么此时有关于AP状态处理,可以直接由STA状态机进行,实现STA状态和AP状态的共存。

[0058] 在上述实施例的基础上,在所述转入与所述接口类型相对应的状态机之后,所述方法还包括:

[0059] 保持与所述接口类型不一致的状态机处于开启状态。

[0060] 由上述实施例的内容可知,本发明实施例将与应用层对接的WifiController状态机分解为了只处理AP状态的WifiAPController状态机与只处理STA状态的WifiSTAController状态机,在处理接口类型为AP状态时,直接转入WifiAPController状态机即可,在处理接口类型为STA状态时,直接转入WifiSTAController状态机即可。

[0061] 可以理解的是,本发明实施例分解之后的WifiAPController状态机和WifiSTAController状态机是独立运行的,且本发明实施例需要实现同时对AP状态和STA状态进行处理,出于对系统资源的充分利用那么优选的,本发明实施例在转入其中一个状态机后,另一类的状态机无需进行关闭操作,从而节省了系统资源。具体的,WifiAPController状态机和WifiSTAController状态机同时保持开启状态,从而当任意类型的状态接入时,都能直接根据其类型转入相应的WifiController状态机中,即AP状态接入时直接转入WifiAPController状态机,STA状态接入时直接转入WifiSTAController状态机,两类状态机同时开启互不影响,各自处理对应的状态。

[0062] 图6是本发明实施例提供的一种在Android系统中支持AP和STA共存的装置结构示意图,如图6所示,包括:应用层对接模块601以及驱动层对接模块602,其中:

[0063] 应用层对接模块601用于将WifiController状态机分解为控制AP状态的状态机以及控制STA状态的状态机,并根据所述Android系统应用层调用开启的接口类型,转入与所述接口类型相对应的状态机,所述接口类型包括AP状态或STA状态;

[0064] 驱动层对接模块602用于将WifiStateMachine状态机中所有AP状态剥离,并使剥离后的所述WifiStateMachine状态机中每个STA状态监听所有被剥离的AP状态。

[0065] 具体的如何通过应用层对接模块601以及驱动层对接模块602可用于执行图1所示的在Android系统中支持AP和STA共存的方法实施例的技术方案,其实现原理和技术效果类似,此处不再赘述。

[0066] 本发明实施例提供的一种在Android系统中支持AP和STA共存的装置,通过将Android系统框架层中的Wi-Fi状态机分解成两部分,分别独立连接处理应用程序和驱动层的指令,每一部分都能保障AP和STA的独立运行,避免由于Android框架层中的Wi-Fi状态机的错乱。

[0067] 在上述实施例的基础上,所述驱动层对接模块具体用于:

[0068] 将所述WifiStateMachine状态机中所有AP状态剥离并存储在内存中,并将内存地址分配给剥离后的WifiStateMachine状态机中每个STA状态,以使剥离后的WifiStateMachine状态机中每个STA状态监听所述内存。

[0069] 在上述实施例的基础上,所述装置还包括:

[0070] AP状态存储模块,用于若所述接口状态为AP状态,则将当前AP状态值保存在所述内存中。

[0071] 在上述实施例的基础上,所述装置还包括:

[0072] STA状态处理模块,用于若所述接口类型为STA状态,则直接转入剥离AP状态后的WifiStateMachine状态机。

[0073] 在上述实施例的基础上,所述装置还用于:

[0074] 保持与所述接口类型不一致的状态机处于开启状态。

[0075] 图7是本发明实施例提供的一种电子设备的结构框图,参照图7,所述电子设备,包括:处理器(processor)701、通信接口(Communications Interface)702、存储器(memory)703和总线704,其中,处理器701,通信接口702,存储器703通过总线704完成相互间的通信。处理器701可以调用存储器703中的逻辑指令,以执行如下方法:将WifiController状态机分解为控制AP状态的状态机以及控制STA状态的状态机,并根据所述Android系统应用层调用开启的接口类型,转入与所述接口类型相对应的状态机,所述接口类型包括AP状态或STA状态;将WifiStateMachine状态机中所有AP状态剥离,并使剥离后的所述WifiStateMachine状态机中每个STA状态监听所有被剥离的AP状态。

[0076] 本发明实施例公开一种计算机程序产品,所述计算机程序产品包括存储在非暂态计算机可读存储介质上的计算机程序,所述计算机程序包括程序指令,当所述程序指令被计算机执行时,计算机能够执行上述各方法实施例所提供的方法,例如包括:将WifiController状态机分解为控制AP状态的状态机以及控制STA状态的状态机,并根据所述Android系统应用层调用开启的接口类型,转入与所述接口类型相对应的状态机,所述接口类型包括AP状态或STA状态;将WifiStateMachine状态机中所有AP状态剥离,并使剥离后的所述WifiStateMachine状态机中每个STA状态监听所有被剥离的AP状态。

[0077] 本发明实施例提供一种非暂态计算机可读存储介质,所述非暂态计算机可读存储介质存储计算机指令,所述计算机指令使所述计算机执行上述各方法实施例所提供的方法,例如包括:将WifiController状态机分解为控制AP状态的状态机以及控制STA状态的状态机,并根据所述Android系统应用层调用开启的接口类型,转入与所述接口类型相对应的状态机,所述接口类型包括AP状态或STA状态;将WifiStateMachine状态机中所有AP状态剥离,并使剥离后的所述WifiStateMachine状态机中每个STA状态监听所有被剥离的AP状态。

[0078] 通过以上的实施方式的描述,本领域的技术人员可以清楚地了解到各实施方式可借助软件加必需的通用硬件平台的方式来实现,当然也可以通过硬件。基于这样的理解,上述技术方案本质上或者说对现有技术做出贡献的部分可以以软件产品的形式体现出来,该计算机软件产品可以存储在计算机可读存储介质中,如ROM/RAM、磁碟、光盘等,包括若干指令用以使得一台计算机设备(可以是个人计算机,服务器,或者网络设备等)执行每个实施例或者实施例的某些部分所述的方法。

[0079] 最后应说明的是:以上实施例仅用以说明本发明的技术方案,而非对其限制;尽管参照前述实施例对本发明进行了详细的说明,本领域的普通技术人员应当理解:其依然可以对前述各实施例所记载的技术方案进行修改,或者对其中部分技术特征进行等同替换;而这些修改或者替换,并不使相应技术方案的本质脱离本发明各实施例技术方案的精神和范围。

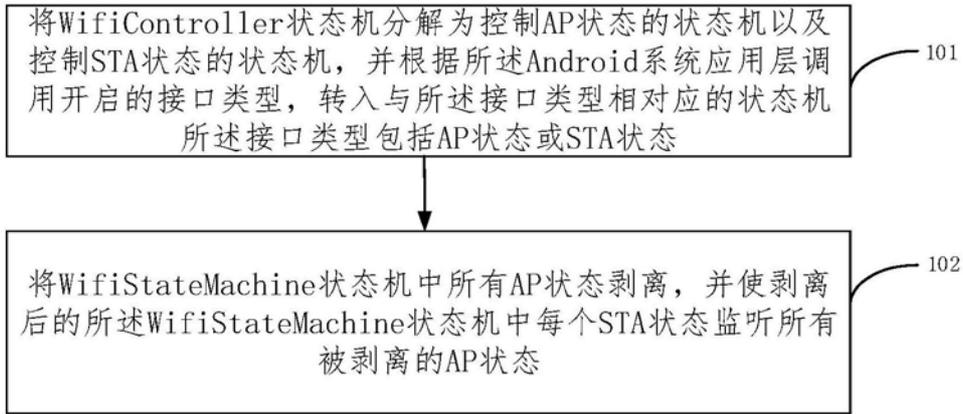


图1

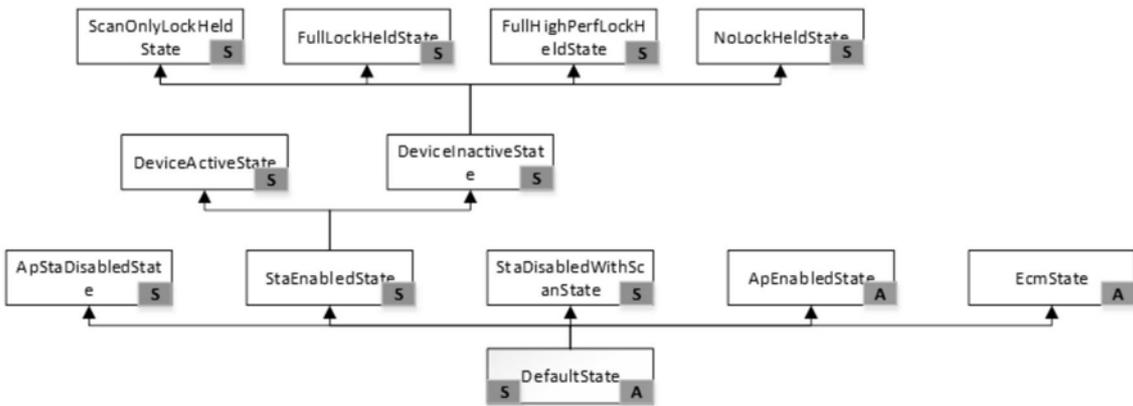


图2

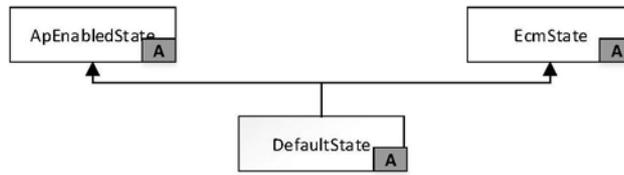
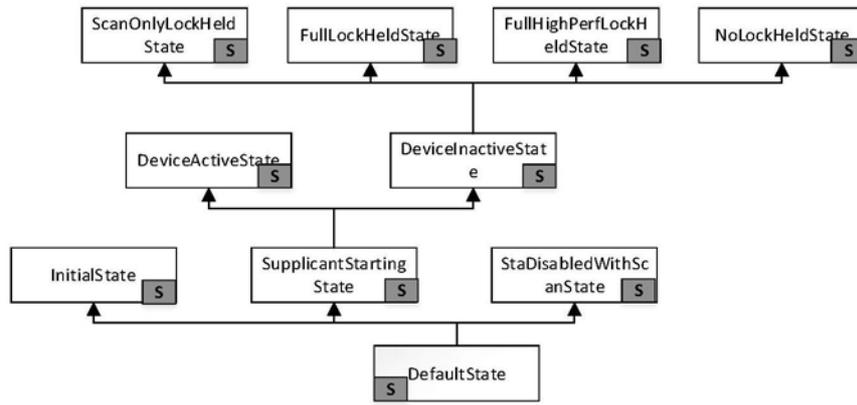


图3

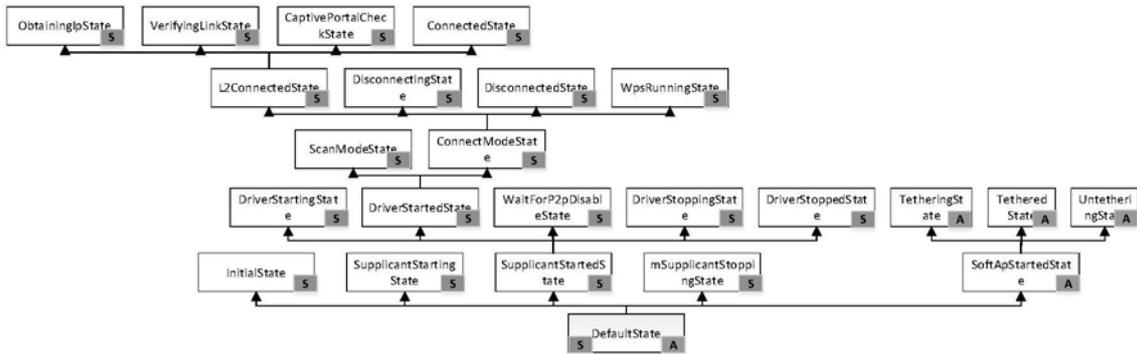


图4

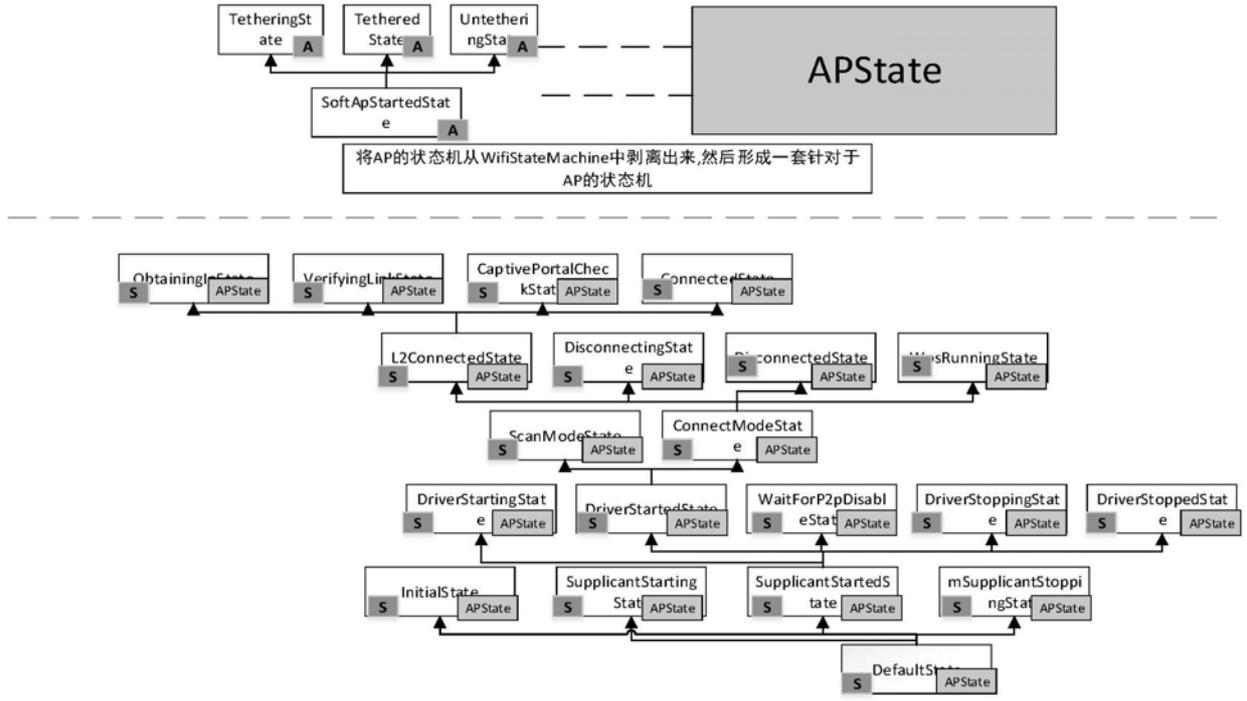


图5

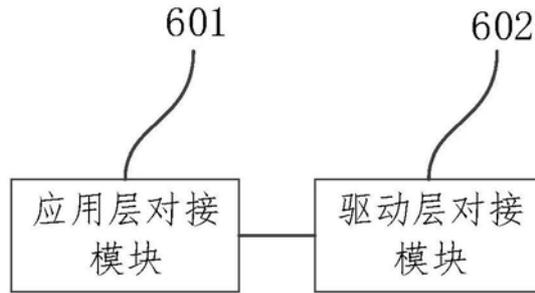


图6

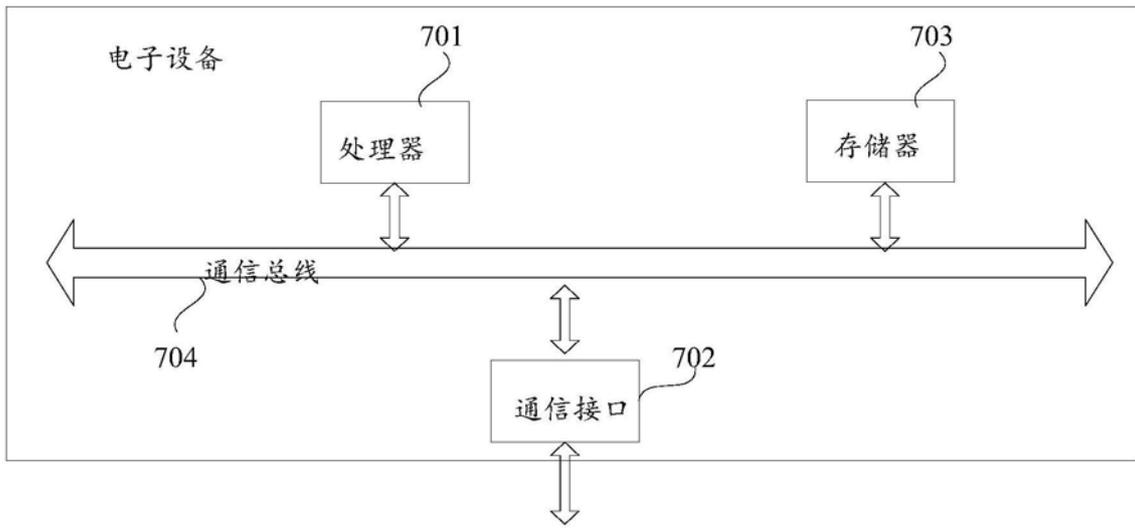


图7