(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
30 November 2017 (30.11.2017)

WIPO | PCT

(10) International Publication Number
WO 2017/204822 A1

(51) International Patent Classification:
G06F 21/57 (2013.01) *G06F 21/60* (2013.01)

(21) International Application Number:
PCT/US2016/034631

(22) International Filing Date:
27 May 2016 (27.05.2016)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: **HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.** [US/US]; 11445 Compaq Center Drive West, Houston, Texas 77070 (US).

(72) Inventors: **NELSON, Marvin D,**; 11311 Chinden Blvd, Boise, Idaho 83714 (US). **MESA, Honee L,**; 11311 Chinden Blvd, Boise, Idaho 83714 (US). **JERAN, Paul**; 11311 Chinden Blvd, Boise, Idaho 83714 (US). **GUNNING, Chris R,**; 11311 Chinden Blvd, Boise, Idaho 83714 (US). **NESS, Erik D,**; Columbia Tech Center, 1115 SE 164th Ave, Columbia Center, Suite 210, Vancouver, Washington 98683 (US).

(74) Agent: **LEMMON, Marcus**; HP Inc, 3390 E. Harmony Road, Mail Stop 35, Fort Collins, Colorado 80528 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:
— *as to the identity of the inventor (Rule 4.17(i))*

Published:
— *with international search report (Art. 21(3))*

(54) Title: FIRMWARE MODULE ENCRYPTION

(57) Abstract: Examples associated with firmware encryption are described. One example device firmware includes a base module. The base module controls a base function of the device. The device firmware also includes a first encrypted module that modifies a first function of the device. The first encrypted module is inactive until decrypted. A decryption module decrypts the first module using a first encryption key and controls activation of the first encrypted module.

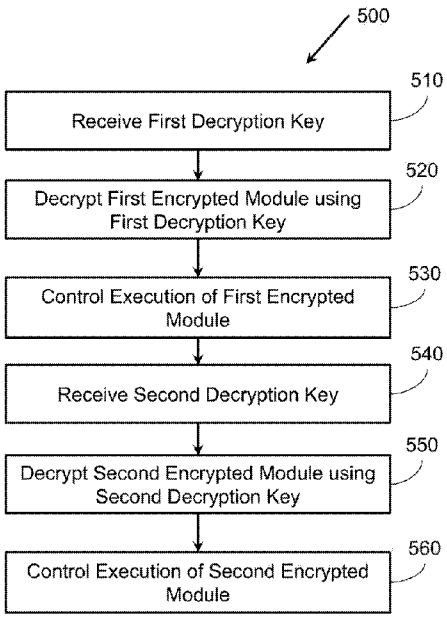

Figure 5

WO 2017/204822 A1

FIRMWARE MODULE ENCRYPTION

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] The subject matter of this application is related to that of copending patent application Ser. No. _____ filed concurrently herewith by M. Nelson et al. for PRINTER FIRMWARE ENCRYPTION (Atty. Docket No. 84460072) and assigned to a common assignee. The disclosure of application Ser. No. _____ is incorporated herein by reference.

[0002] The matter of this application is related to that of copending patent application Ser. No. _____ filed concurrently herewith by M. Nelson et al. for PRINTER AUTHENTICATION (Atty. Docket No. 84460076) and assigned to a common assignee. The disclosure of application Ser. No. _____ is incorporated herein by reference.

BACKGROUND

[0003] A device firmware is a set of instructions embedded in the device that facilitate controlling, monitoring, and so forth, the device and/or or components of the device. In various examples, the device firmware may be held in a non-volatile memory and may rarely, if ever, be changed during the life cycle of the device (depending on the type of device). In some devices, the firmware may be the primary enabler of device functionality. In other devices, the firmware may act as an interface between device hardware and applications installed on the device. Devices that include various types of firmware may include, for example, personal computers, printers, other peripherals, appliances, remote controls, digital watches, cellular phones, digital cameras, and so forth.

BRIEF DESCRIPTION OF THE DRAWINGS

[0004] The present application may be more fully appreciated in connection with the following detailed description taken in conjunction with the accompanying drawings.

[0005] FIG. 1 illustrates an example device firmware associated with firmware encryption.

[0006] FIG. 2 illustrates an example device associated with firmware encryption.

[0007] FIG. 3 illustrates another example device associated with firmware encryption.

[0008] FIG. 4 illustrates a flowchart of example operations associated with firmware encryption.

[0009] FIG. 5 illustrates another flowchart of example operations associated with firmware encryption.

[0010] FIG. 6 illustrates another flowchart of example operations associated with firmware encryption.

[0011] FIG. 7 illustrates an example computing device in which example systems, and methods, and equivalents, may operate.

DETAILED DESCRIPTION

[0012] Devices, methods, and equivalents associated with firmware encryption are described. As discussed above, device firmware facilitates control, monitoring, and so forth of device functionality. In some cases, it may be desirable to activate, release, or otherwise modify a device functionality after a certain event in a life cycle of a device. By way of illustration, consider a device feature that is not ready to be activated at a release date of the device, but uses specialized firmware. Because updating firmware of the device may be difficult (e.g., if firmware is embedded in read

only memory), it may be desirable to embed the specialized firmware in the device in a manner that allows later activation of the firmware. In other examples, device security may depend in part on security of the device firmware, and changing the device firmware periodically may make it more difficulty to attack the device. Instead of updating the firmware, it may be possible to embed updated security measures in the firmware that can be activated periodically over time. In some situations, it may be desirable to encrypt these modules to keep their behavior undiscoverable and/or unknown prior to their decryption. Additionally, encrypting modules may also facilitate reducing the number of persons with the ability to access the modules prior to the production and/or release of the devices into which the modules are embedded.

[0013] Consequently, device firmware may include firmware that controls base device functionality, as well as a series of encrypted firmware modules that otherwise update, modify, enhance, disable, replace, add to, and so forth, the base device functionality. The different encrypted firmware modules may be encrypted using different encryption keys. When a decryption module, also embedded in the device firmware, receives encryption keys, respective encrypted firmware modules may be decrypted and activated. This may allow updating device functionality, activating device functionality, updating device security, and so forth.

[0014] Figure 1 illustrates an example device firmware 100 associated with firmware encryption. It should be appreciated that the items depicted in figure 1 are illustrative examples, and many different systems, devices, and so forth, may operate in accordance with various examples.

[0015] Figure 1 illustrates an example device firmware 100 associated with firmware encryption. Device firmware 100 includes a base module 110. Base firmware module 110 may control a variety of device functions 180 associated with a device into which device firmware 100 is embedded. By way of illustration the device into which device firmware 100 is embedded may be, for example, a computer, a printer, an appliance, manufacturing equipment, a mobile device, and so forth. Consequently, device functions 180 may vary depending on what type of device into which device firmware 100 is embedded. By way of illustration, a printer may have device functions 180 that relate to printing, scanning, copying, dialing fax numbers,

emailing, performing device maintenance, connecting to other devices, communicating information to and receiving information from users, and so forth. These device functions 180 may be implemented in software, hardware, firmware, and so forth, as well as combinations thereof. For example, a printing function may use software instructions to convert a file into a printable format. The printable format may be interpreted and an interface in firmware may be used to control the printer hardware to physically cause a hard copy of the file to be generated by transferring print material (e.g., ink) from a print container to a print media (e.g., paper).

[0016] Thus, base module 110 may be firmware that controls operations of a device into which device firmware 100 is inserted. In some examples, for less sophisticated devices, the firmware may be all that is used to control operation of a device into which device firmware 100 is inserted. By way of illustration, a remote control may primarily operate based on firmware that causes specific signals to be transmitted by the remote upon certain presses of buttons on the remote. A device receiving the signals may do more work, possibly involving firmware, hardware, software, and so forth, to accomplish a task based on the signal received from the remote.

[0017] In other examples, base modules 110 may operate as an interface between applications and device functions 180. In examples where device firmware 100 is embedded in a more sophisticated device such as a personal computer or mobile device, many of the applications interfacing with base module 110 may be executing on the device in which device firmware 100 resides. In other examples, base module 110 may interface with applications external to the device in which device firmware 100 is embedded. Other examples of roles performed by base module 110 in controlling device functions 180 may also be possible.

[0018] Device firmware 100 also contains a set of encrypted modules including a first encrypted module 120 and a second encrypted module 130. Though two encrypted modules are illustrated, device firmware 100 may include numerous encrypted modules that perform a variety of functions when activated. The encrypted modules may be embedded into device firmware 100 at the same time as base module 110 with the intent that the encrypted modules be activated independent from base

module 110. In various examples described herein, the encrypted modules may be configured such that they do not operate without being decrypted independently. By way of illustration, some devices employ firmware encryption techniques to hinder malicious attacks against the device. In examples described herein, the encrypted modules may operate differently than these firmware encryption techniques because they are decrypted separately than the remainder of the firmware (e.g., base module 110). This allows the decrypted modules to, for example, reside inactive in firmware until activated by being decrypted. This may, for example, facilitate device security, allow late delivery of functionality to the device, and so forth.

[0019] Consequently, device firmware 100 may contain a decryption module 140 for the purpose of decrypting the encrypted modules. Decryption module 140, upon obtaining a decryption key from a key provider 199, may decrypt an encrypted module to which the decryption key corresponds. Upon decryption of this encrypted module, the encrypted module may activate. Key provider 199 may be, for example, built into a device into which device firmware 100 is embedded. In this example, key provider 199 may be a secure application specific integrated circuit. In other examples, key provider 199 may be external to the device into which device firmware 100 is embedded. When this device is network connected, key provider 199 may be a manufacturer or other type of service provider for the device. Other key providers may include, for example, users, other devices similar to and/or connected to the device in which device firmware 100 is embedded, trusted third parties, and so forth. Whether key provider 199 is internal or external to the device may depend on, for example, security concerns, a likelihood device firmware 100 will have access to a network connection, how critical the encrypted modules will be to device functionality, and so forth. By way of illustration, for certain products, it may be undesirable to mandate that the product be periodically connected to a network. In these examples, the key provider may reside within products. For other devices, where device security is important, it may be possible to remotely provide decryption keys for encrypted modules via a network.

[0020] In some examples, decrypted modules may modify existing device functions 180. In this example first encrypted module 120 is indicated as being

affecting an existing device function 180. In various examples, first encrypted module 120 may modify, upgrade, replace, deactivate, and so forth an existing device function 180 when first encrypted module 120 is decrypted and activated. By way of illustration, a printer may use a variety of print consumables. Some of the print consumables may be containers of print material (e.g., ink, toner, a 3D print material). To hinder consumption of counterfeit print containers by the printer, the printer may authenticate print containers. Thus, encrypted modules may be embedded in the firmware of the printer that use a variety of authentication techniques over time. This may cause the printer to obtain changing credentials from print containers over time, making it more difficult to manufacture counterfeit print containers that keep up with the changing credentials. Embedding the encrypted modules in the firmware of the printer may allow the printer to automatically adjust authentication techniques over time, without requiring a firmware update to be received over a network connection or installed by a user. That the encrypted modules are not received or installed after product distribution may be additionally valuable because their activation can be made non-optional. While a user could opt not to install a firmware update, firmware modules that are capable of being automatically decrypted upon receiving a decryption key may facilitate managing future behavior of a product to that controls behavior of the device without user interaction. This may include for example, disabling features no longer under contract, enhancing security, and so forth.

[0021] In other examples, decrypted modules may add additional functionality 185 to a device into which device firmware 100 is embedded. By way of illustration, a fitness wearable may be designed with a GPS functionality that is not quite ready at the release of the wearable. In this example, the firmware may be ready, but the applications that use the firmware may be finished after users have begun using the wearable. Consequently, firmware associated with the GPS functionality may be disabled by encrypting the firmware until an appropriate decryption key is received by the wearable. This may allow the GPS functionality to be added without updating the firmware after release of the wearable. Using an encrypted module instead of merely disabling the feature may allow the code to remain undiscoverable, thereby hindering undesired activation by an enterprising user. This may allow the device manufacturer

to control when the additional functionality is activated and ensure they have an initial opportunity to provide the functionality to the customer.

[0022] In various examples, decryption keys may be provided by key provider 199 to decryption module 140 on a set schedule. The schedule may be defined prior to release of the device into which device firmware 100 is embedded. In some examples, the set schedule may be based on specific dates, passage of time following activation of the device into which device firmware 100 is embedded, and so forth. In other examples, the set schedule may be based on usage of the device into which device firmware 100 is embedded, consumption of components or supplies by the device, and so forth. Releasing a key based on a usage or consumption based scenario may facilitate, for example, maintaining a device into which device firmware 100 is embedded, rewarding a user of the device, and so forth. In various examples, the criteria and/or schedule for releasing decryption keys may be protected from discovery or alteration using embedded security hardware, encryption and/or signing technologies, and so forth.

[0023] In other examples, the decryption keys may be provided without a set schedule. For example, key provider 199 may provide a decryption key to decryption module 140 to decrypt second encrypted module to modify a device function 180 or activate additional functionality 185 after key provider 199 receives a payment from a user. In a similar example, a user may act as key provider 199 themselves after obtaining a key. Examples where keys are provided without a known schedule may be appropriate to encourage behavior from a user, allow trial functionality of device features, release functionality to a user on a subscription basis, and so forth.

[0024] In addition to features discussed above, firmware encryption may facilitate enhanced organizational control over information related to development of devices. During the process of manufacturing devices, many individuals may have access to device firmware including developers, manufacturers, device testers, and so forth. Each additional person with access to device firmware may pose an additional risk of leaking important information to competitors, counterfeiters, and/or other individuals with malicious intent (e.g., hackers). By way of illustration, printer manufacturers often begin seeing counterfeit supplies appear on the market within

weeks of a product release, a feat that may only be achievable with aid of leaked information. Encrypting firmware may allow an organization to restrict knowledge regarding device behaviors to a limited number of individuals, thereby reducing a risk of leaking valuable corporate information. To illustrate, device behavior could be changed on launch day of a product by triggering decryption of an encrypted firmware module, thereby reducing the value certain information obtained prior to the launch day.

[0025] To further achieve the goal of reducing access to firmware and/or device functionality prior to its activation, various components of device firmware 100 and/or the device into which device firmware 100 is embedded may be generated and or installed into the device using secure manufacturing processes. These manufacturing processes may, for example, programmatically manipulate firmware modules so that modules in separate devices are made up of differing instructions that cause similar results. In other examples, release schedules, authentication materials, and so forth may be manipulated and or otherwise securely embedded into the device via its components (e.g., a secure ASIC that releases encryption keys), components of device firmware 100, and so forth.

[0026] It is appreciated that, in the following description, numerous specific details are set forth to provide a thorough understanding of the examples. However, it is appreciated that the examples may be practiced without limitation to these specific details. In other instances, methods and structures may not be described in detail to avoid unnecessarily obscuring the description of the examples. Also, the examples may be used in combination with each other.

[0027] "Module", as used herein, includes but is not limited to instructions stored on a computer-readable medium or in execution on a machine that perform a function(s) or an action(s), and/or to cause a function or action from another module, method, and/or system. Where multiple logical modules are described, it may be possible to incorporate the multiple logical modules into one logical module. Similarly, where a single logical module is described, it may be possible to distribute that single logical module between multiple logical modules.

[0028] Figure 2 illustrates an example device 200 associated with firmware encryption. Device 200 includes a device firmware 210. Device firmware 210 includes a base module 220. Base module 220 may control a base function of device 200. The base function of device 200 may be one of numerous functions 250 of device 200. The base function, as well as other functions of device 200 may be performed by hardware, software, firmware, other components, and/or a combination of components of device 200. Activities associated with device functions 250 may depend in part on what functions device 200 is designed to perform. By way of illustration, if device 200 is a printer, device functions may relate to, for example, printing, scanning, copying, cleaning print heads, other maintenance tasks, managing components of the printer, managing supplies and/or containers of supplies consumed by the printer, communicating with external devices (e.g., personal computers), and so forth. A fitness wearable may have firmware associated with, for example, communicating with nearby devices, motion tracking, displaying data to a user, heartrate monitoring, and so forth. Firmware associated with a remote control may simply control what signal is transmitted (e.g., via an infrared medium) when certain buttons are pressed.

[0029] Device firmware 210 also includes a first encrypted module 230. The first encrypted module may be inactive until decrypted. This may, for example, hinder undesired analysis of the first encrypted module prior to its activation, prevent early use of a functionality associated with the first encrypted module, and so forth. The first encrypted module may modify a first function of device 200. As with the base function controlled by base module 220, the first function may be one of numerous device functions 250 performed by device 200. Modifying the first function may include, for example, activating the first function, replacing executable instructions associated with the first function, changing a way the first function is performed, changing a component of device 200 performing the first function, deactivating the first function, and so forth. When activating a device function 250, first encrypted module 230 may provide executable instructions to device 200 that cause device 200 to perform the first function. In various examples, decryption of first encrypted module 220 and modifications to the first function, including activation of the first function, may

occur during operation of device 200. Thus, a reset or power cycle of device 200 may be unnecessary to decrypt and execute first encrypted module 230.

[0030] In some examples, the base function and the first function may be the same function. Consequently, the first encrypted module may modify the base function. In some examples, once decrypted, first encrypted module 230 may prevent further operation of base module 220, thereby causing first encrypted module 230 to replace base module 220 during operation of device 200.

[0031] Device firmware 210 also includes a decryption module 240. First decryption module 240 may decrypt first encrypted module 230 using a first encryption key. The encryption key may be received from, for example, another component of device 200, an external device via a network connection, an external device over a direct connection, a user input to device 200, and so forth. In some examples, the encryption key may be obtained from multiple sources and assembled by decryption module 240.

[0032] Figure 3 illustrates a device 300 associated with firmware encryption. Device 300 includes several items similar to those described above with reference to device 200 (figure 2). For example, device 300 includes a device firmware 310 containing a base module 320, a first encrypted module 330, and a decryption module 340. The base module and the first encrypted module 330 may affect operation of a set of device functions 350.

[0033] Device firmware 310 also includes a second encrypted module 335. Second encrypted module 335 may modify a second function of device 300. In some examples, the second function of device 300 may be one of the numerous device functions 350 performed by device 300, and may be the same function as a base function controlled by base module 320 and/or a first function affected by first encrypted module 330. Second encrypted module 335 may be inactive until decrypted by decryption module 340. Decryption module 340 may use a second encryption key to decrypt second encrypted module 335.

[0034] Device 300 also includes an application specific integrated circuit (ASIC) 360. ASIC 360 may securely store encryption keys including a first encryption key used to decrypt first encrypted module 330 and the second encryption key. The encryption keys stored in ASIC 360 may be periodically provided by ASIC 360 to decryption module 340 causing decryption module 340 to decrypt corresponding encryption modules of device firmware 310.

[0035] Though two encrypted modules are illustrated, device 300 may include numerous encrypted modules. Each encrypted module may have a corresponding decryption key that causes decryption module 340 to decrypt and activate respective encrypted modules. These encrypted modules may be installed in device 300 to reduce scenarios requiring updating firmware of device 300, but still allowing updates to the device firmware. Further, as the updates are encrypted, it may be difficult for a person attempting to maliciously affect device 300 to identify countermeasures built into encrypted modules before the encrypted modules are activated.

[0036] Figure 4 illustrates an example method 400 associated with firmware encryption. Method 400 may be embodied on a non-transitory processor-readable medium storing processor-executable instructions. The instructions, when executed by a processor, may cause the processor to perform method 400.

[0037] Method 400 includes receiving a first decryption key at 410. The decryption key may be received in the firmware of a device. The decryption key may be associated with a first encrypted module. The first encrypted module may be embedded in the firmware of the device. The first encrypted module may modify a function of the device. Modifying a function of a device may include, for example, activating the function, disabling the function, changing how the function operates, and so forth.

[0038] Method 400 also includes decrypting the first encrypted module at 420. The first encrypted module may be decrypted using the first decryption key. Method 400 also includes controlling execution of the first encrypted module at 430. Execution of the first encrypted module may occur after completing decryption of the first encrypted module. Consequently, method 400 may illustrate how a device may

activate an encrypted module embedded within the device. This may allow the device to, for example, securely update itself without an external source providing an updated firmware image.

[0039] Figure 5 illustrates a method 500 associated with firmware decryption. Method 500 includes several actions similar to those described above with reference to method 400 (figure 4). For example, method 500 includes receiving a first decryption key at 510, decrypting a first encrypted module at 520, and controlling execution of the first encrypted module at 530.

[0040] Method 500 also includes receiving a second decryption key at 540. The second decryption key may be associated with a second encrypted module. The second encrypted module may be embedded in the firmware of the device.

[0041] Method 500 also includes decrypting the second encrypted module using the second encryption key at 550. The second encrypted module may be decrypted using the second decryption key. Method 500 also includes controlling execution of the second encrypted module at 560. The second encrypted module may be executed upon completing decryption of the second encrypted module.

[0042] Figure 6 illustrates a method 600 associated with firmware encryption. Method 600 includes embedding a series of encrypted modules in the firmware of a device at 610. The series of encrypted modules may be scheduled to be decrypted on a set schedule. The set schedule may be, for example, a temporal schedule, a usage based schedule, a maintenance schedule, and so forth. A temporal schedule may be based on, for example, specific dates and times, passage of time after an initial activation of the device, and so forth. A usage based schedule may be based on, for example, how often the device is used, how much the device consumes a resource (e.g., a printer's consumption of ink), and so forth. A maintenance schedule may be based on, for example, when certain maintenance events have occurred, when certain maintenance events are expected to occur based on, device usage, and so forth. By way of illustration, a device may have certain wear and tear over time, and updating the device firmware to mitigate the wear and tear at specific points in time based on usage of the device may be desirable.

[0043] Method 600 also includes embedding a decryption module in the firmware of the device at 620. The decryption module may receive decryption keys associated with encrypted modules. The decryption module may use the decryption keys to decrypt corresponding encrypted modules. The decryption module may also control execution of the decrypted modules. In some examples, upon decryption of a member of the series of encrypted modules, a previous member of the series of encrypted modules may be deactivated. In other examples, decrypted modules may remain functional over the remaining life cycle of the device.

[0044] Method 600 also includes controlling delivery of the decryption keys to the decryption module at 630. In some examples, controlling delivery of the decryption keys may be achieved by embedding a secure delivery vector into the device. Consequently, the secure delivery vector may provide the decryption keys to the decryption module in association with the set schedule. In other examples, delivery of the decryption keys may occur by providing the keys to the device over a network, providing the keys to a user who inputs the keys into the device, and so forth.

[0045] In various examples, components embedded into the device during their respective actions may be embedded during manufacturing of the device using a secure process. The secure process may facilitate updating and/or modifying the components between their design and when the components are embedded into the device in a manner that facilitates reducing access to specific release details of the components. By way of illustration, during design, a placeholder release date may be used that is modified prior to manufacturing of the device by an administrator based on a confidential planned release schedule for the feature. In other examples, embedding components into the device using a secure process may facilitate modifying authentication materials, and so forth.

[0046] Figure 7 illustrates an example device in which example systems and methods, and equivalents, may operate. The example device may be a device 700 that includes a processor 710 and a memory 720 connected by a bus 730. Device 700 includes a firmware encryption module 740. Firmware encryption module 740 may perform, alone or in combination, various functions described above with

reference to the example devices, methods, and so forth. In different examples, firmware encryption module 740 may be implemented as a non-transitory computer-readable medium storing processor-executable instructions.

[0047] The instructions may also be presented to device 700 as data 750 and/or process 760 that are temporarily stored in memory 720 and then executed by processor 710. The processor 710 may be a variety of processors including dual microprocessor and other multi-processor architectures. Memory 720 may include non-volatile memory (e.g., read only memory) and/or volatile memory (e.g., random access memory). Memory 720 may also be, for example, a magnetic disk drive, a solid state disk drive, a floppy disk drive, a tape drive, a flash memory card, an optical disk, and so on. Thus, memory 720 may store process 760 and/or data 750. Device 700 may also be associated with other devices including other computers, devices, peripherals, and so forth in numerous configurations (not shown).

[0048] It is appreciated that the previous description of the disclosed examples is provided to enable any person skilled in the art to make or use the present disclosure. Various modifications to these examples will be readily apparent to those skilled in the art, and the generic principles defined herein may be applied to other examples without departing from the spirit or scope of the disclosure. Thus, the present disclosure is not intended to be limited to the examples shown herein but is to be accorded the widest scope consistent with the principles and novel features disclosed herein.

WHAT IS CLAIMED IS:

1. A device firmware, comprising:

a base module to control a base function of the device;

a first encrypted module to modify a first function of the device, where the first encrypted module is inactive until decrypted; and

a decryption module to decrypt the first encrypted module using a first encryption key and to control activation of the first encrypted module.

2. The device firmware of claim 1, comprising a second encrypted module to modify a second function of the device, where the second encrypted module is inactive until decrypted, and where the decryption module decrypts the second encrypted module using a second encryption key.

3. The device firmware of claim 1, where the base function and the first function are the same function.

4. The device firmware of claim 1, where the first encryption key is received from one of an application specific integrated circuit built into the device, another component of the device, an external device over a network connection, an external device over a direct connection, and a user input.

5. The device firmware of claim 1, where the decryption module assembles the encryption key from pieces of the encryption key obtained from multiple sources.

6. The device firmware of claim 1, where the first encrypted module replaces the base module during operation of the device.

7. The device firmware of claim 1, where the first encrypted module affects how the device interacts with a consumable supply container.

8. The device firmware of claim 1, where modifying the first function of the device includes activating the first function of the device, and where activating the first function of the device includes providing executable instructions to the device that cause the device to perform the first function.

9. The device firmware of claim 8, where the first function of the device is activated during operation of the device.

10. A method, comprising:

receiving, in the firmware of a device, a first decryption key associated with a first encrypted module embedded in the firmware of the device;

decrypting the first encrypted module using the first decryption key; and

controlling execution of the first encrypted module upon completing decryption of the first encrypted module.

11. The method of claim 10, comprising:

receiving a second decryption key associated with a second encrypted module embedded in the firmware of the device;

decrypting the second encrypted module using the second decryption key; and

controlling execution of the second encrypted module upon completing decryption of the second encrypted module.

12. A method, comprising:

embedding, in the firmware of a device, a series of encrypted modules to be decrypted on a set schedule;

embedding, in the firmware of the device, a decryption module to receive decryption keys associated with the encrypted modules, to decrypt encrypted modules using respective decryption keys, and to control execution of the encrypted modules; and

controlling delivery of the decryption keys to the decryption module.

13. The method of claim 11, where, upon decryption of a member of the series of encrypted modules, a previous member of the series of encrypted modules is deactivated.

14. The method of claim 11, where controlling delivery of the encryption keys to the decryption module includes embedding a secure delivery vector into the device, where the secure delivery vector provides the decryption keys to the decryption module in association with the set schedule.

15. The method of claim 14, where at least one of a member of the series of encrypted modules, the decryption module, and the secure delivery vector are embedded during manufacturing of the device using a secure process.

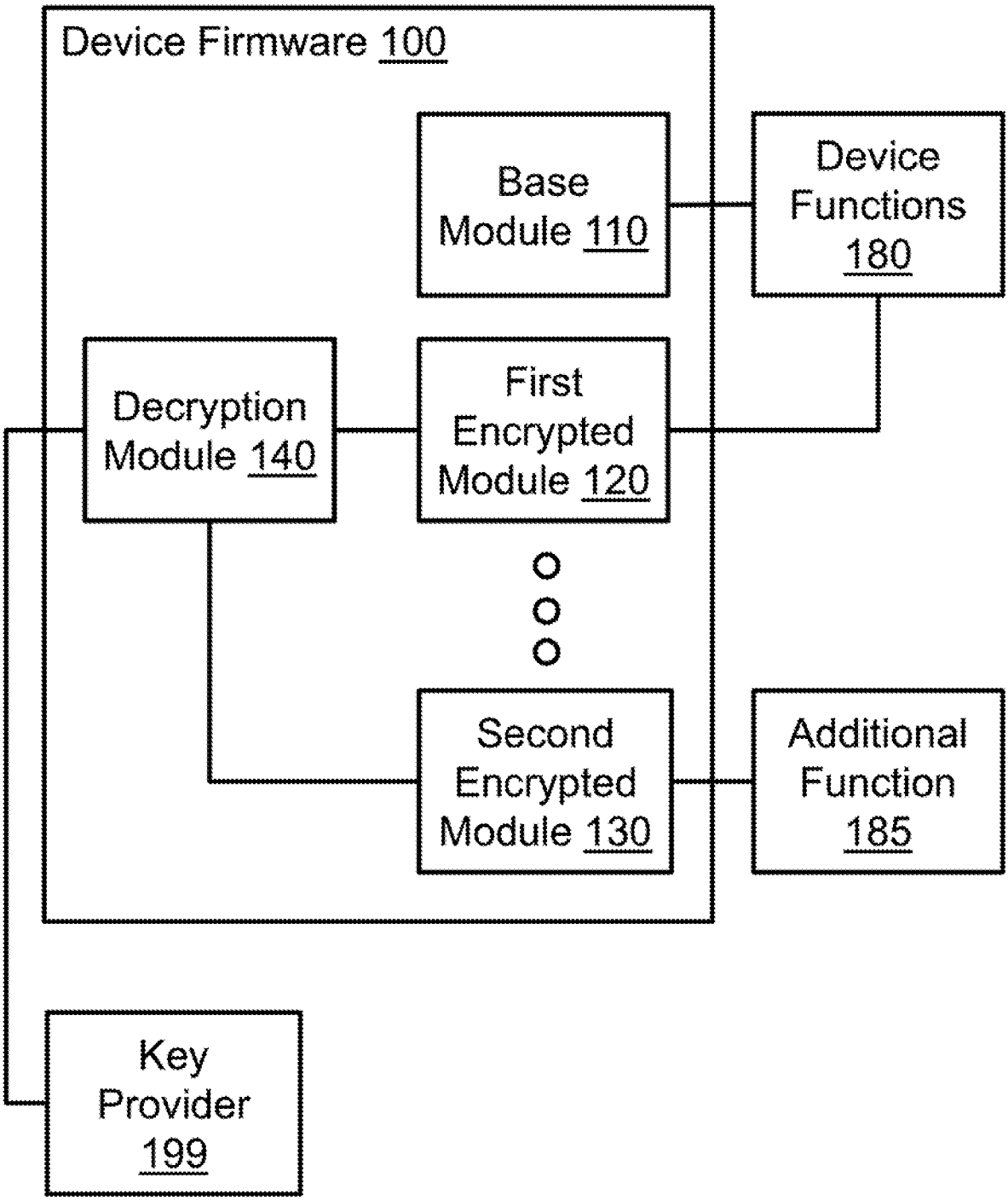
Device Firmware 100
Base Module 110
Device Functions 180
Decryption Module 140
First Encrypted Module 120
Second Encrypted Module 130
Additional Function 185
Key Provider 199

Figure 1

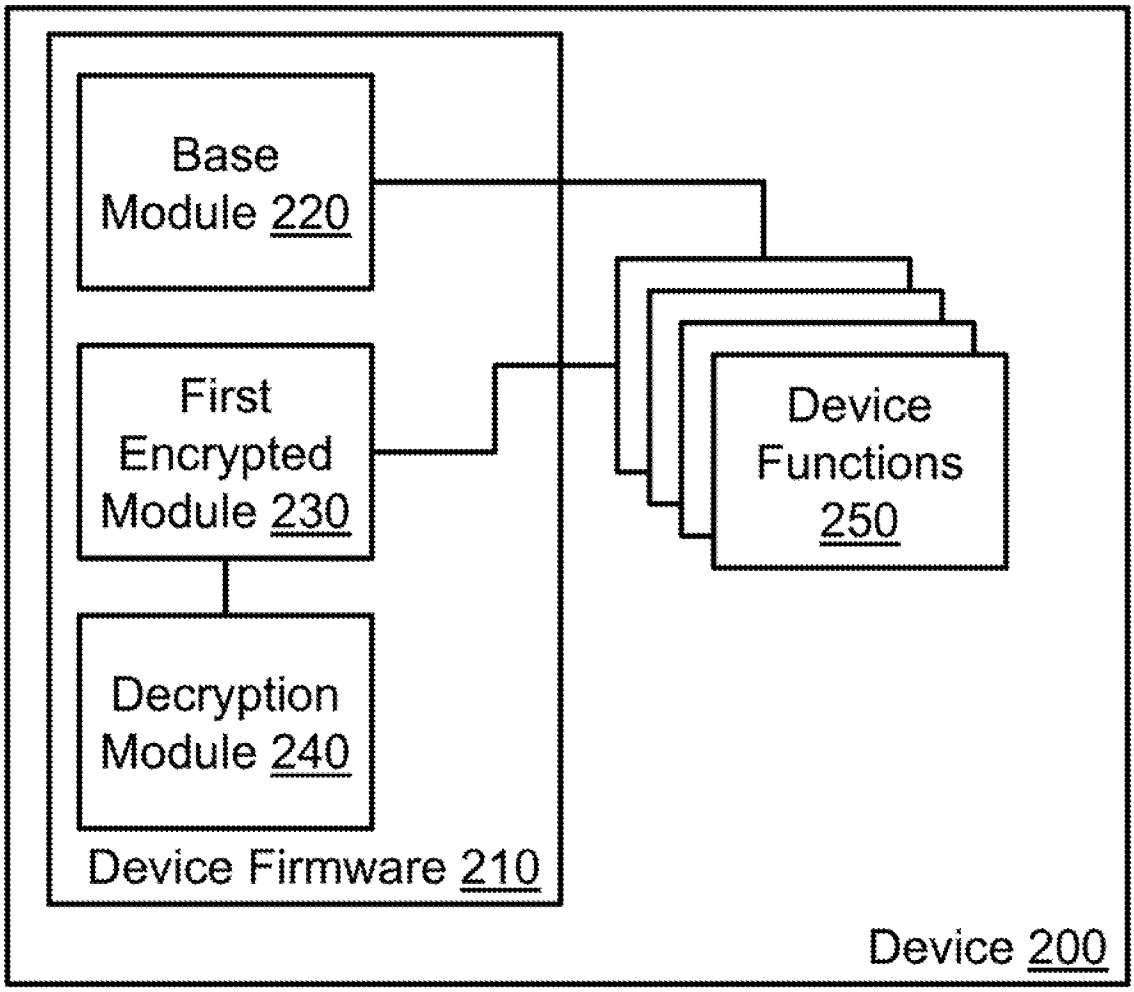
Base Module 220
First Encrypted Module 230
Decryption Module 240
Device Firmware 210
Device Functions 250
Device 200

Figure 2

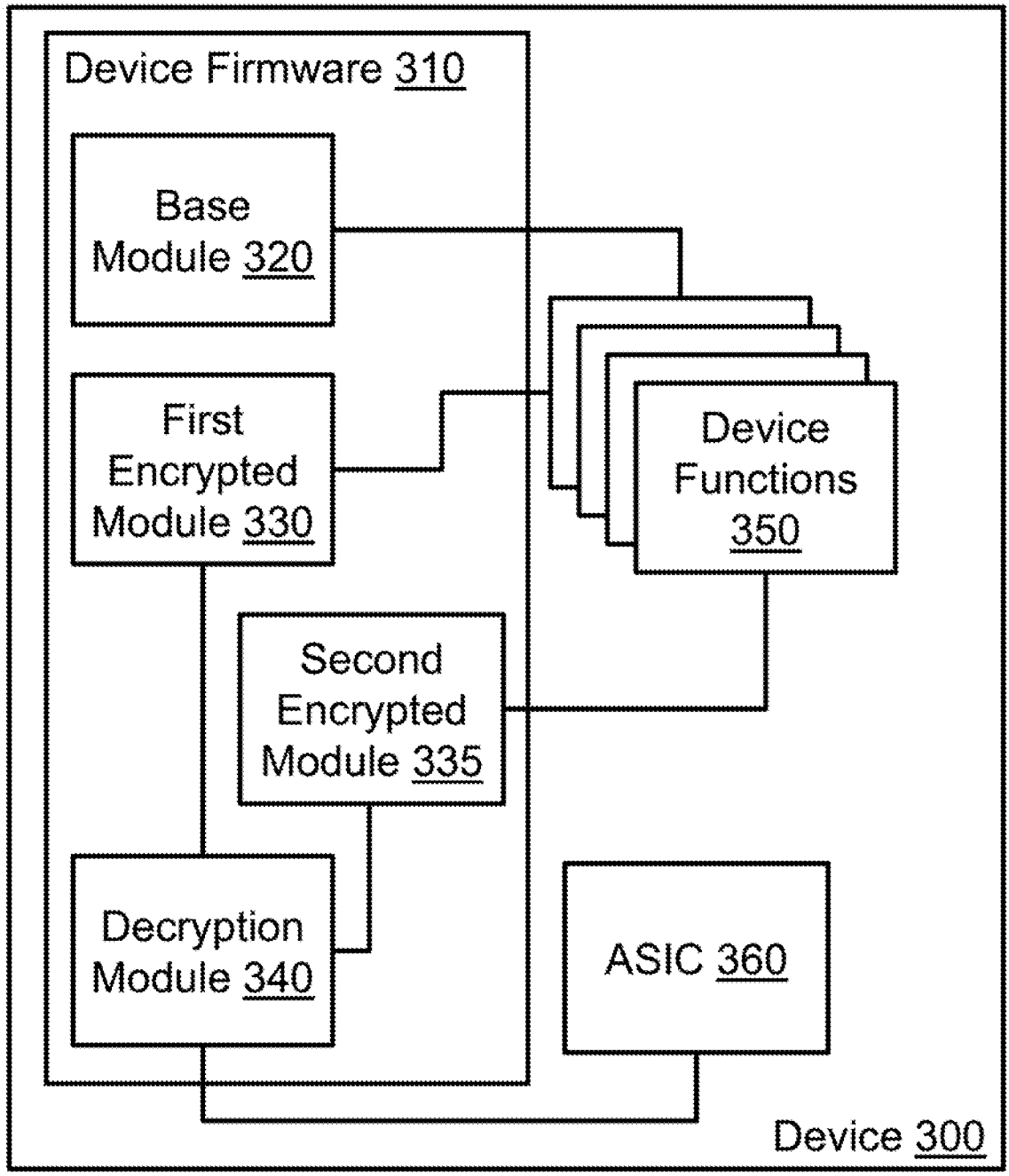
Device Firmware 310
Base Module 320
First Encrypted Module 330
Second Encrypted Module 335
Decryption Module 340
Device Functions 350
ASIC 360
Device 300

Figure 3

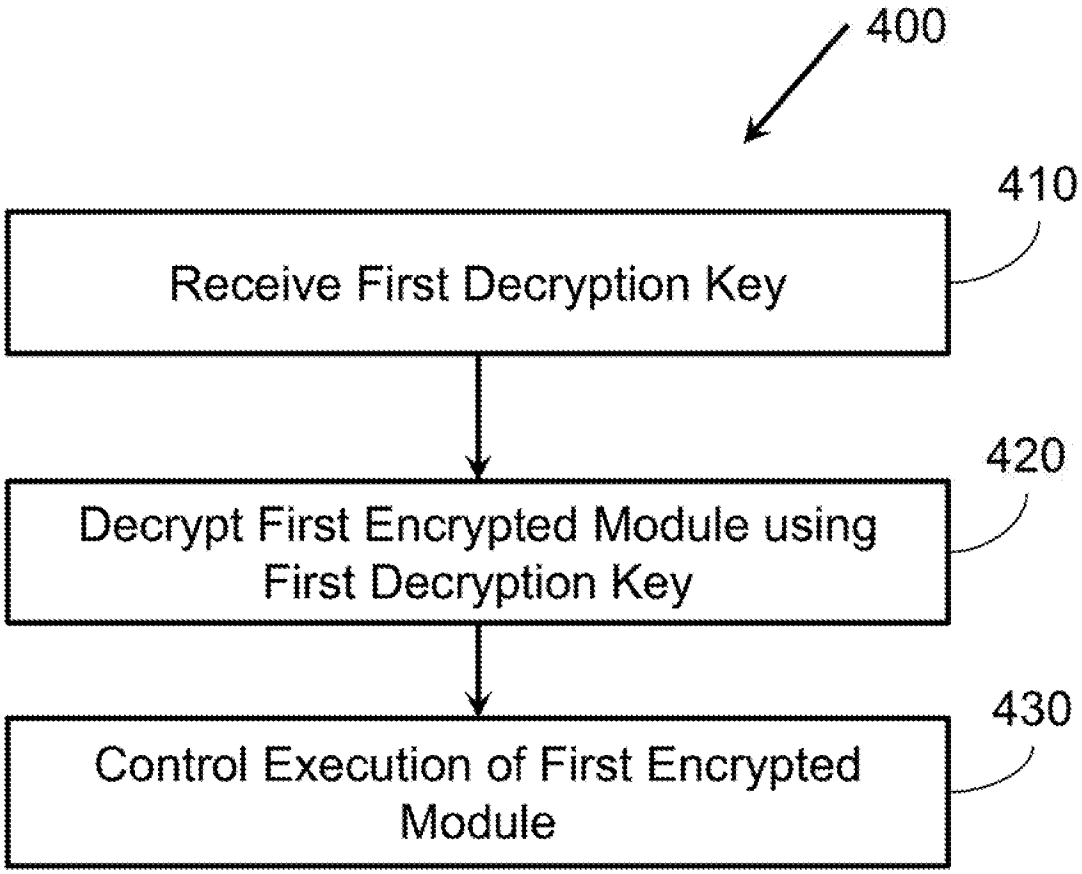
400
410
Receive First Decryption Key
420
Decrypt First Encrypted Module using First Decryption Key
430
Control Execution of First Encrypted Module

Figure 4

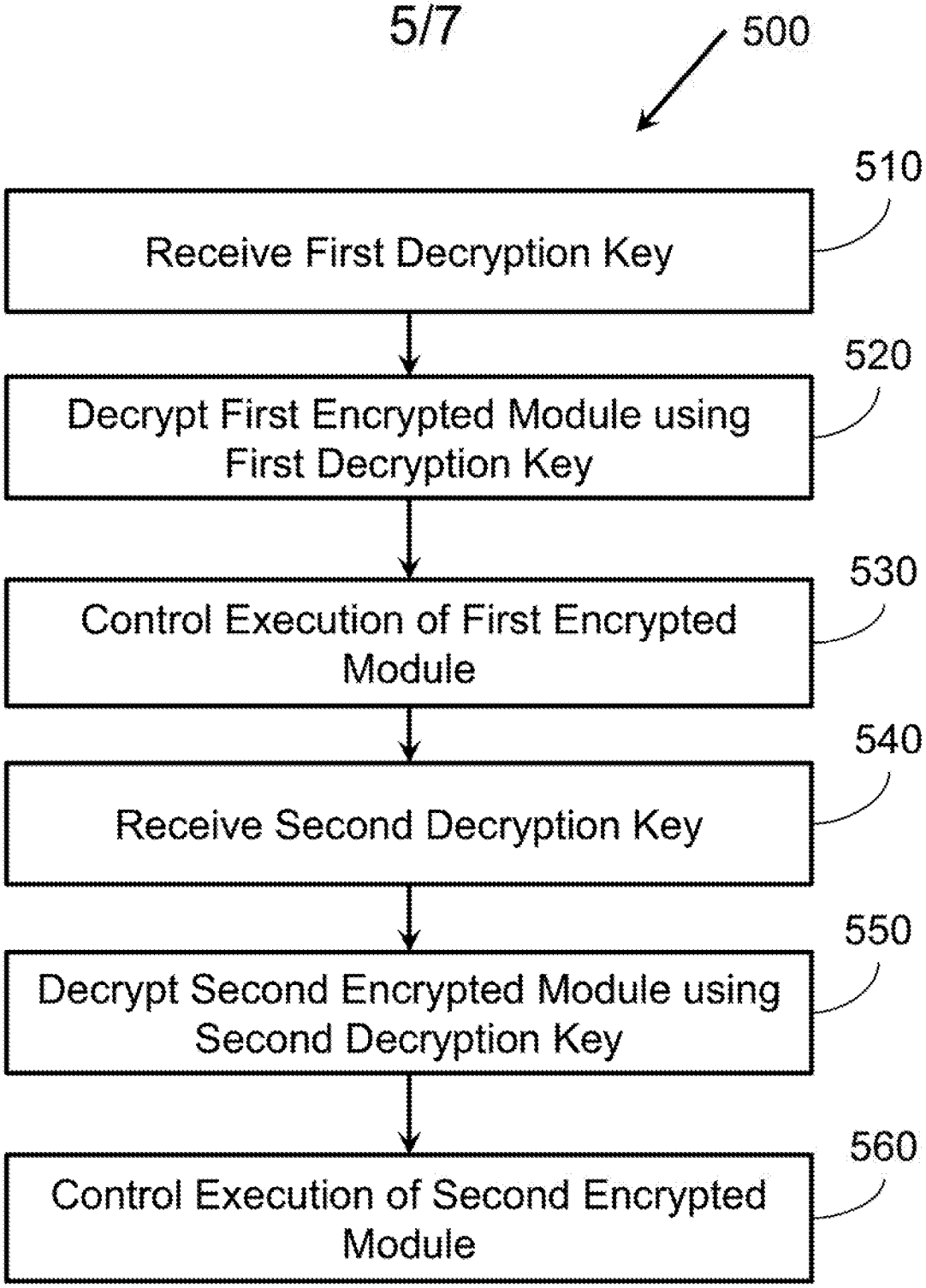
500
510
Receive First Decryption Key
520
Decrypt First Encrypted Module using First Decryption Key
530
Control Execution of First Encrypted Module
540
Receive Second Decryption Key
550
Decrypt Second Encrypted Module using Second Decryption Key
560
Control Execution of Second Encrypted Module

Figure 5

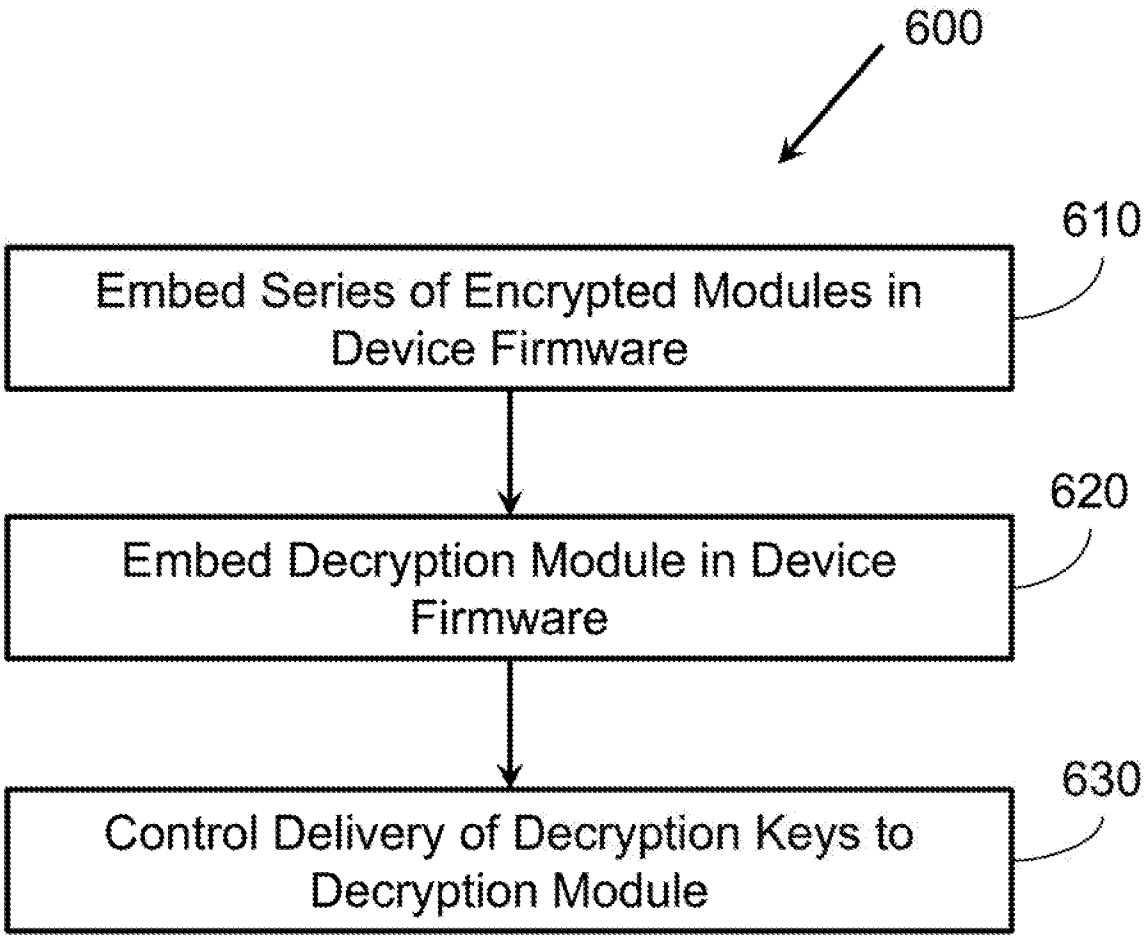
600
610
Embed Series of Encrypted Modules in Device Firmware
620
Embed Decryption Module in Device Firmware
630
Control Delivery of Decryption Keys to Decryption Module

Figure 6

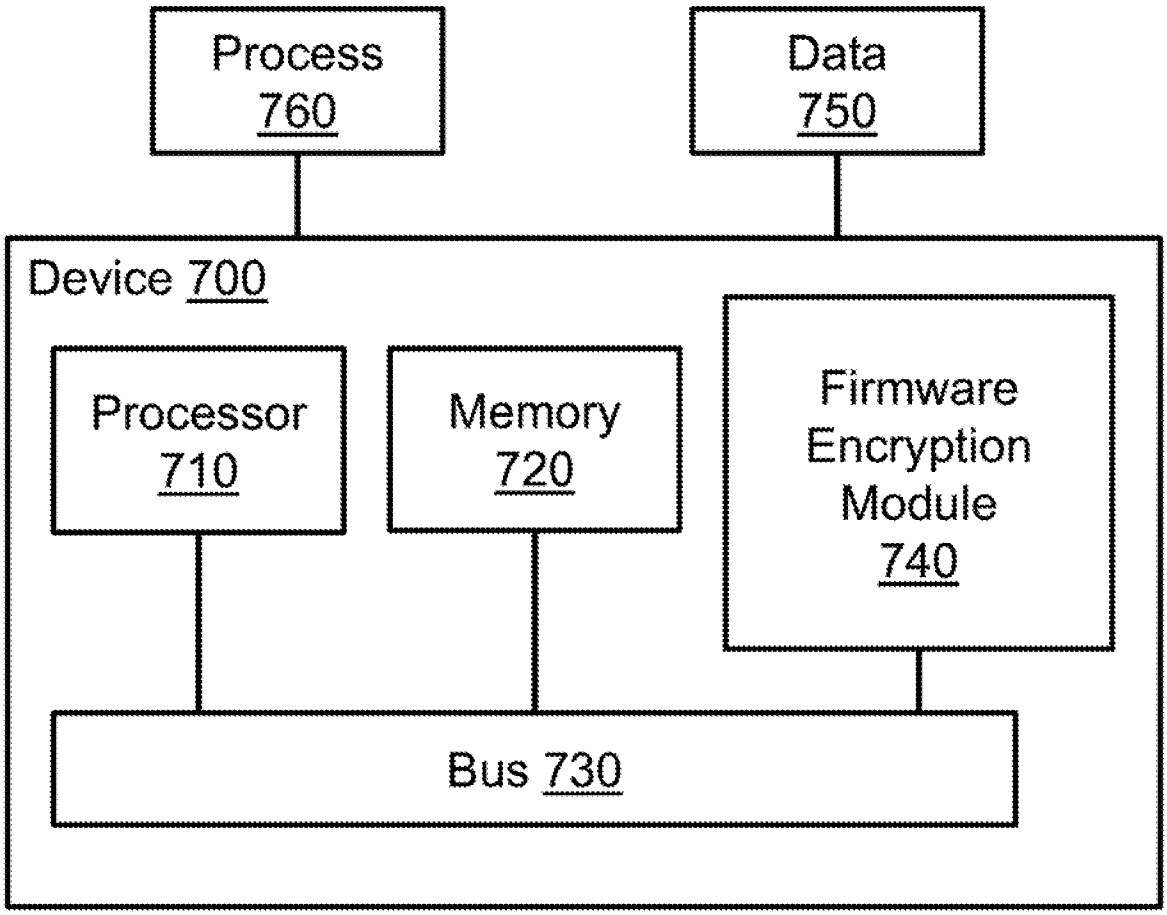
Process
760
Data
750
Device 700
Processor
710
Memory
720
Firmware
Encryption
Module
740
Bus 730

Figure 7

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2016/034631

A. CLASSIFICATION OF SUBJECT MATTER

G06F 21/57(2013.01)i, G06F 21/60(2013.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06F 21/57; H04L 9/08; G06F 12/14; G06F 21/02; G06Q 20/00; G03G 15/00; G06F 21/00; G06F 21/60

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Korean utility models and applications for utility models
Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
eKOMPASS(KIPO internal) & Keywords: firmware, encryption, decryption, key, module, modify, function, control, ASIC

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2007-0088613 A1 (CHRISTOPHER A. ADKINS et al.) 19 April 2007 See paragraphs [0006], [0017], [0020], [0039], [0044]-[0046], [0050], [0071]; and figure 1.	1-15
Y	US 2014-0169803 A1 (IKE SEUNG HO LEE) 19 June 2014 See paragraphs [0010], [0028]-[0029], [0033]; and figures 1-3.	1-15
A	US 2004-0034785 A1 (HORNG-MING TAI et al.) 19 February 2004 See paragraphs [0008], [0046]; and figure 1.	1-15
A	US 2010-0174913 A1 (SIMON B. JOHNSON et al.) 08 July 2010 See paragraphs [0013], [0027]; and figure 1.	1-15
A	EP 1892641 A2 (STMICROELECTRONICS, INC) 27 February 2008 See paragraphs [0003], [0021]; and figure 1.	1-15

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents,such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
06 February 2017 (06.02.2017)	**07 February 2017 (07.02.2017)**

Name and mailing address of the ISA/KR
International Application Division
Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea
Facsimile No. +82-42-481-8578

Authorized officer
CHIN, Sang Bum
Telephone No. +82-42-481-8398

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2016/034631

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2007-0088613 A1	19/04/2007	EP 1949258 A2	30/07/2008
		EP 1949258 A4	03/08/2011
		WO 2007-047500 A2	26/04/2007
		WO 2007-047500 A3	15/11/2007
US 2014-0169803 A1	19/06/2014	None	
US 2004-0034785 A1	19/02/2004	None	
US 2010-0174913 A1	08/07/2010	US 2010-0174922 A1	08/07/2010
		US 9286493 B2	15/03/2016
EP 1892641 A2	27/02/2008	EP 1892641 A3	25/02/2009
		US 2008-0052518 A1	28/02/2008