CLAIMS

1. A method for a modem-directed application processor boot sequence in a portable computing device (PCD), the method comprising:

initializing a direct memory access (DMA) engine for the boot sequence, wherein the DMA engine is configured to read data from a memory component;

initializing a crypto engine for the boot sequence, wherein the crypto engine is configured to calculate a hash according to a predetermined hash function;

reading with the DMA engine metadata and data segments associated with an image stored in the memory component and associated with the boot sequence;

calculating with the crypto engine hash values associated with the metadata and data segments; and

transitioning an application processor into an idle state for durations of time coinciding with at least one of reading with the DMA engine during the boot sequence and calculating with the crypto engine during the boot sequence.

2. The method of claim 1, wherein the memory component is a non-volatile flash memory component.

3. The method of claim 1, wherein the application processor is a heterogeneous, multi-core processor.

4. The method of claim 1, wherein the idle state is a wait-for-interrupt (WFI) idle state.

5. The method of claim 1, wherein the PCD comprises at least one of a cellular telephone, a satellite telephone, a personal digital assistant (PDA), a smartphone, a navigation device, a tablet, a smartbook, a media player and a laptop computer.

6. A method for a modem-directed application processor boot flow in a portable computing device (PCD), the method comprising:

a. initializing, with an application processor in an active processing state, a direct memory access (DMA) engine, wherein the DMA engine is configured to read data from a memory component;

b. initializing, with the application processor, a crypto engine, wherein the crypto engine is configured to calculate a hash according to a predetermined hash function;

c. reading, with the DMA engine, metadata associated with an image stored in the memory component;

d. transitioning the application processor from the active processing state into an idle state;

e. calculating, with the crypto engine, a hash value associated with the metadata;

f. transitioning the application processor out of the idle state to the active processing state;

g. authenticating, with the application processor, the metadata by validating the hash value and a signature associated with the metadata;

h. reading, with the DMA engine, a current segment of the image;

i. transitioning the application processor from the active processing state to the idle state;

j. calculating, with the crypto engine, a hash value associated with the current segment of the image;

k. transitioning the application processor out of the idle state to the active processing state;

l. reading, with the DMA engine, a next segment of the image;

m. authenticating, with the application processor, the current segment of the image by validating the hash value and a signature associated with the current segment of the image;

n. transitioning the application processor from the active processing state to the idle state;

o. calculating, with the crypto engine, a hash value associated with the next segment of the image; and

p. repeating actions k through o until a last segment of the image is authenticated and then bringing a subsystem out of reset.

7. The method of claim 6, wherein the memory component is a non-volatile flash memory component.

8. The method of claim 6, wherein the application processor is a heterogeneous, multi-core processor.

9. The method of claim 6, wherein the idle state is a wait-for-interrupt (WFI) idle state.

10. The method of claim 6, wherein the PCD comprises at least one of a cellular telephone, a satellite telephone, a personal digital assistant (PDA), a smartphone, a navigation device, a tablet, a smartbook, a media player and a laptop computer.

11. A computer system for modem-directed application processor boot sequence in a portable computing device (PCD), the system comprising:

an application processor configured to:

initialize a direct memory access (DMA) engine for the boot sequence, wherein the DMA engine is configured to read data from a memory component;

initialize a crypto engine for the boot sequence, wherein the crypto engine is configured to calculate a hash according to a predetermined hash function;

instruct the DMA engine to read metadata and data segments associated with an image stored in the memory component and associated with the boot sequence;

instruct the crypto engine to calculate hash values associated with the metadata and data segments; and

transition into an idle state for durations of time coinciding with at least one of reading by the DMA engine and calculating by the crypto engine.

12. The system of claim 11, wherein the memory component is a non-volatile flash memory component.

13. The system of claim 11, wherein the application processor is a heterogeneous, multi-core processor.

14. The system of claim 11, wherein the idle state is a wait-for-interrupt (WFI) idle state.

15. The system of claim 11, wherein the PCD comprises at least one of a cellular telephone, a satellite telephone, a personal digital assistant (PDA), a smartphone, a navigation device, a tablet, a smartbook, a media player and a laptop computer.

16. A computer system for modem-directed application processor boot flow in a portable computing device (PCD), the system comprising:

an application processor configured to:

a. initialize, with an application processor in an active processing state, a direct memory access (DMA) engine, wherein the DMA engine is configured to read data from a memory component;

b. initialize, with the application processor, a crypto engine, wherein the crypto engine is configured to calculate a hash according to a predetermined hash function;

c. instruct, with the application processor, the DMA engine to read metadata associated with an image stored in the memory component;

d. instruct, with the application processor, the crypto engine to calculate a hash value associated with the metadata;

e. transition the application processor from the active processing state into an idle state;

f. calculate, with the crypto engine, the hash value associated with the metadata;

g. transition the application processor out of the idle state to the active processing state;

h authenticate, with the application processor, the metadata by validating the hash value and a signature associated with the metadata;

i. instruct, with the application processor, the DMA engine to read a current segment of the image;

j. instruct, with the application processor, the crypto engine to calculate a hash value associated with the current segment of the image;

k. transition the application processor from the active processing state to the idle state;

l. calculate, with the crypto engine, a hash value associated with the current segment of the image;

m. transition the application processor out of the idle state to the active processing state;

n. instruct, with the application processor, the DMA engine to read a next segment of the image;

o. authenticate, with the application processor, the current segment of the image by validating the hash value and a signature associated with the current segment of the image;

p. instruct, with the application processor, the crypto engine to calculate a hash value associated with the next segment of the image;

q. transition the application processor from the active processing state into the idle state;

r. calculate, with the crypto engine, the hash value associated with the next segment of the image; and

s. repeat actions m through r until a last segment of the image is authenticated and then bring a subsystem out of reset.

17. The system of claim 16, wherein the memory component is a non-volatile flash memory component.

18. The system of claim 16, wherein the application processor is a heterogeneous, multi-core processor.

19. The system of claim 16, wherein the idle state is a wait-for-interrupt (WFI) idle state.

20. The system of claim 16, wherein the PCD comprises at least one of a cellular telephone, a satellite telephone, a personal digital assistant (PDA), a smartphone, a navigation device, a tablet, a smartbook, a media player and a laptop computer.

21. A computer system for a modem-directed application processor boot sequence in a portable computing device (PCD), the system comprising:

means for initializing a direct memory access (DMA) engine for the boot sequence, wherein the DMA engine is configured to read data from a memory component;

means for initializing a crypto engine for the boot sequence, wherein the crypto engine is configured to calculate a hash according to a predetermined hash function;

means for instructing the DMA engine to read metadata and data segments associated with an image stored in the memory component and associated with the boot sequence;

means for instructing the crypto engine to calculate hash values associated with the metadata and data segments; and

means for transitioning an application processor into an idle state for durations of time coinciding with at least one of reading by the DMA engine the metadata and calculating by the crypto engine during the boot sequence.

22. The computer system of claim 21, wherein the memory component is a non-volatile flash memory component.

23. The computer system of claim 21, wherein the application processor is a heterogeneous, multi-core processor.

24. The computer system of claim 21, wherein the idle state is a wait-for-interrupt (WFI) idle state.

25. The computer system of claim 21, wherein the PCD comprises at least one of a cellular telephone, a satellite telephone, a personal digital assistant (PDA), a smartphone, a navigation device, a tablet, a smartbook, a media player and a laptop computer.

26. A computer program product comprising a non-transitory computer usable medium having a computer readable program code embodied therein, the computer readable program code adapted to be executed to implement a method for a modem-directed application processor boot sequence in a portable computing device (PCD), the method comprising:

initializing a direct memory access (DMA) engine for the boot sequence, wherein the DMA engine is configured to read data from a memory component;

initializing a crypto engine for the boot sequence, wherein the crypto engine is configured to calculate a hash according to a predetermined hash function;

reading with the DMA engine metadata and data segments associated with an image stored in the memory component and associated with the boot sequence;

calculating with the crypto engine hash values associated with the metadata and data segments; and

transitioning an application processor into an idle state for durations of time coinciding with at least one of reading with the DMA engine during the boot sequence and calculating with the crypto engine during the boot sequence.

27. The computer program product of claim 26, wherein the memory component is a non-volatile flash memory component.

28. The computer program product of claim 26, wherein the application processor is a heterogeneous, multi-core processor.

29. The computer program product of claim 26, wherein the idle state is a wait-for-interrupt (WFI) idle state.

30. The computer program product of claim 26, wherein the PCD comprises at least one of a cellular telephone, a satellite telephone, a personal digital assistant (PDA), a smartphone, a navigation device, a tablet, a smartbook, a media player and a laptop computer.

31. A computer program product comprising a non-transitory computer usable medium having a computer readable program code embodied therein, the computer readable program code adapted to be executed to implement a method for a modem-directed application processor boot flow in a portable computing device (PCD), the method comprising:

a. initializing, with an application processor in an active processing state, a direct memory access (DMA) engine, wherein the DMA engine is configured to read data from a memory component;

b. initializing, with the application processor, a crypto engine, wherein the crypto engine is configured to calculate a hash according to a predetermined hash function;

c. reading, with the DMA engine, metadata associated with an image stored in the memory component;

d. transitioning the application processor from the active processing state into an idle state;

e. calculating, with the crypto engine, a hash value associated with the metadata;

f. transitioning the application processor out of the idle state to the active processing state;

g. authenticating, with the application processor, the metadata by validating the hash value and a signature associated with the metadata;

h. reading, with the DMA engine, a current segment of the image;

i. transitioning the application processor from the active processing state to the idle state;

j. calculating, with the crypto engine, a hash value associated with the current segment of the image;

k. transitioning the application processor out of the idle state to the active processing state;

l. reading, with the DMA engine, a next segment of the image;

m. authenticating, with the application processor, the current segment of the image by validating the hash value and a signature associated with the current segment of the image;

n. transitioning the application processor from the active processing state to the idle state;

o. calculating, with the crypto engine, a hash value associated with the next segment of the image; and

p. repeating actions k through o until a last segment of the image is authenticated and then bringing a subsystem out of reset.

32. The computer program product of claim 31, wherein the memory component is a non-volatile flash memory component.

33. The computer program product of claim 31, wherein the application processor is a heterogeneous, multi-core processor.

34. The computer program product of claim 31, wherein the idle state is a wait-for-interrupt (WFI) idle state.

35. The computer program product of claim 31, wherein the PCD comprises at least one of a cellular telephone, a satellite telephone, a personal digital assistant (PDA), a smartphone, a navigation device, a tablet, a smartbook, a media player and a laptop computer.

36. A method for a modem-directed application processor boot sequence in a portable computing device (PCD) and in remote communication with a server, the method comprising:

forming a network of one or more Internet of Things (IoT) devices, the PCD being one of the IoT devices;

a modem of the PCD receiving an instruction from the remote server;

the modem sending a wakeup signal to an application processor to execute a boot sequence;

the application processor initializing a direct memory access (DMA) engine for the boot sequence, wherein the DMA engine is configured to read data from a memory component;

the application processor initializing a crypto engine for the boot sequence, wherein the crypto engine is configured to calculate a hash according to a predetermined hash function;

the DMA engine reading metadata and data segments associated with an image stored in the memory component and associated with the boot sequence;

the crypto engine calculating hash values associated with the metadata and data segments; and

the application processor transitioning into an idle state for durations of time coinciding with at least one of reading with the DMA engine during the boot sequence and calculating with the crypto engine during the boot sequence.

37. The method of claim 36, wherein the memory component is a non-volatile flash memory component.

38. The method of claim 36, wherein the application processor is a heterogeneous, multi-core processor.

39. The method of claim 36, wherein the idle state is a wait-for-interrupt (WFI) idle state.

40. A computer system for modem-directed application processor boot sequence in a portable computing device (PCD), the system comprising:

a remote server in communication with one or more Internet of Things (IoT) devices;

the PCD comprising one of the IoT devices and having a modem configured to receive instructions from the remote server and, in response to the instructions, cause an application processor within the PCD to execute a boot sequence, the application processor configured to:

initialize a direct memory access (DMA) engine for the boot sequence, wherein the DMA engine is configured to read data from a memory component;

initialize a crypto engine for the boot sequence, wherein the crypto engine is configured to calculate a hash according to a predetermined hash function;

instruct the DMA engine to read metadata and data segments associated with an image stored in the memory component and associated with the boot sequence;

instruct the crypto engine to calculate hash values associated with the metadata and data segments; and

transition into an idle state for durations of time coinciding with at least one of reading by the DMA engine and calculating by the crypto engine.

41. The system of claim 40, wherein the memory component is a non-volatile flash memory component.

42. The system of claim 40, wherein the application processor is a heterogeneous, multi-core processor.

43. The system of claim 40, wherein the idle state is a wait-for-interrupt (WFI) idle state.