



US 20230253986A1

(19) **United States**

(12) **Patent Application Publication**

Xin et al.

(10) **Pub. No.: US 2023/0253986 A1**

(43) **Pub. Date: Aug. 10, 2023**

(54) **METHOD FOR AUTO-DETECTION OF WLAN PACKETS USING STF**

(71) Applicant: **Huawei Technologies Co., Ltd.**,
Shenzhen (CN)

(72) Inventors: **Yan Xin**, Kanata (CA); **Sheng Sun**,
Kanata (CA)

(21) Appl. No.: **18/300,907**

(22) Filed: **Apr. 14, 2023**

Related U.S. Application Data

(60) Continuation of application No. 17/028,592, filed on Sep. 22, 2020, now Pat. No. 11,637,572, which is a division of application No. 15/043,089, filed on Feb. 12, 2016, now Pat. No. 10,813,083.

(60) Provisional application No. 62/115,445, filed on Feb. 12, 2015, provisional application No. 62/219,794, filed on Sep. 17, 2015.

Publication Classification

(51) **Int. Cl.**

H03M 13/00 (2006.01)
H04L 27/34 (2006.01)
H04L 27/26 (2006.01)

H04L 69/18 (2006.01)

H04L 27/00 (2006.01)

H04L 1/00 (2006.01)

H04L 5/00 (2006.01)

H04L 69/22 (2006.01)

H04W 72/23 (2006.01)

H04L 27/20 (2006.01)

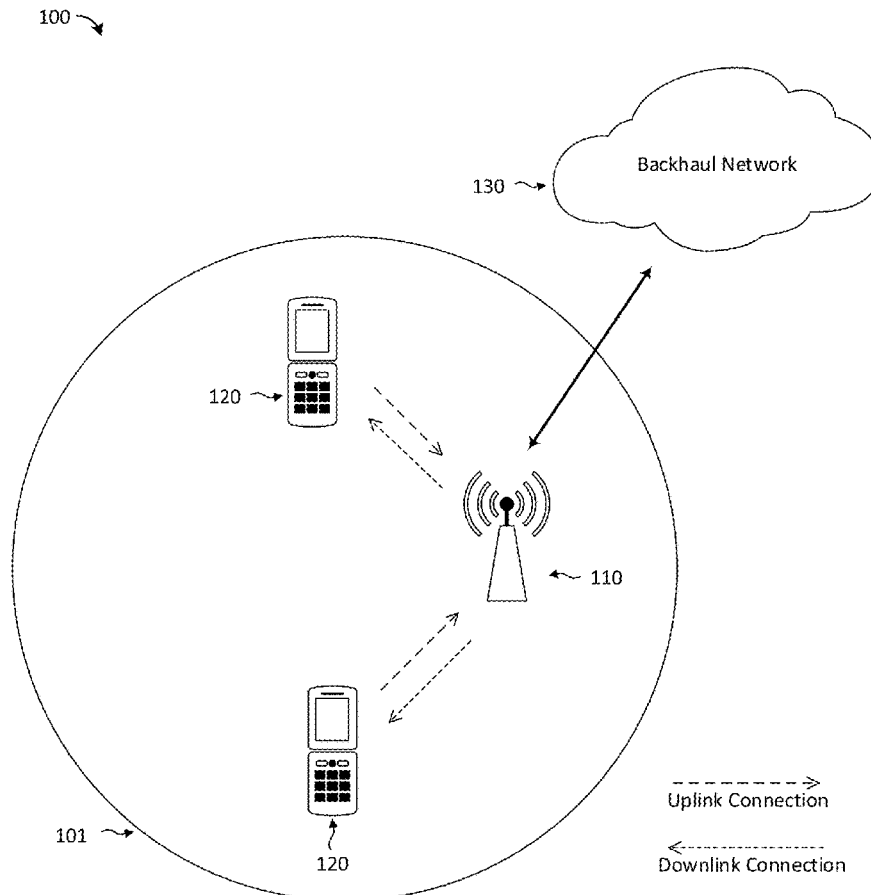
(52) **U.S. Cl.**

CPC **H03M 13/6527** (2013.01); **H04L 27/3455** (2013.01); **H04L 27/2613** (2013.01); **H04L 69/18** (2013.01); **H04L 27/0012** (2013.01); **H04L 1/0091** (2013.01); **H04L 5/0053** (2013.01); **H04L 69/22** (2013.01); **H04W 72/23** (2023.01); **H04L 27/206** (2013.01); **H04L 27/2601** (2013.01); **H03M 13/27** (2013.01)

(57)

ABSTRACT

A method of auto-detection of WLAN packets includes selecting a first Golay sequence from a first pair of Golay complementary sequences associated with first packet type, each Golay sequence of the first pair of Golay complementary sequences being zero correlation zone (ZCZ) sequences with each Golay sequence of a second pair of Golay complementary sequences associated with a second packet type, and transmitting a wireless packet carrying a short training field (STF) that includes one or more instances of the first Golay sequence.



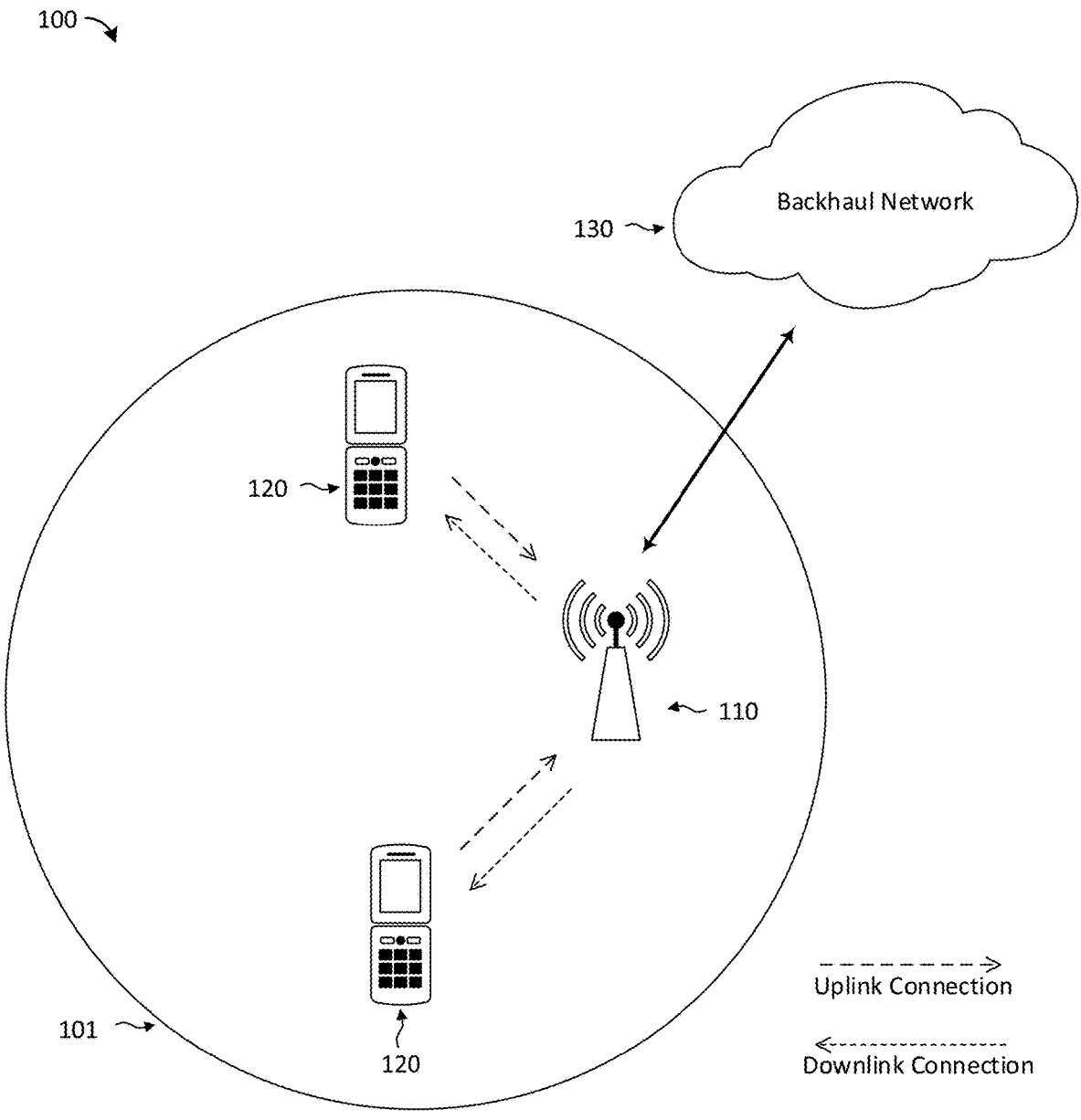


Fig. 1

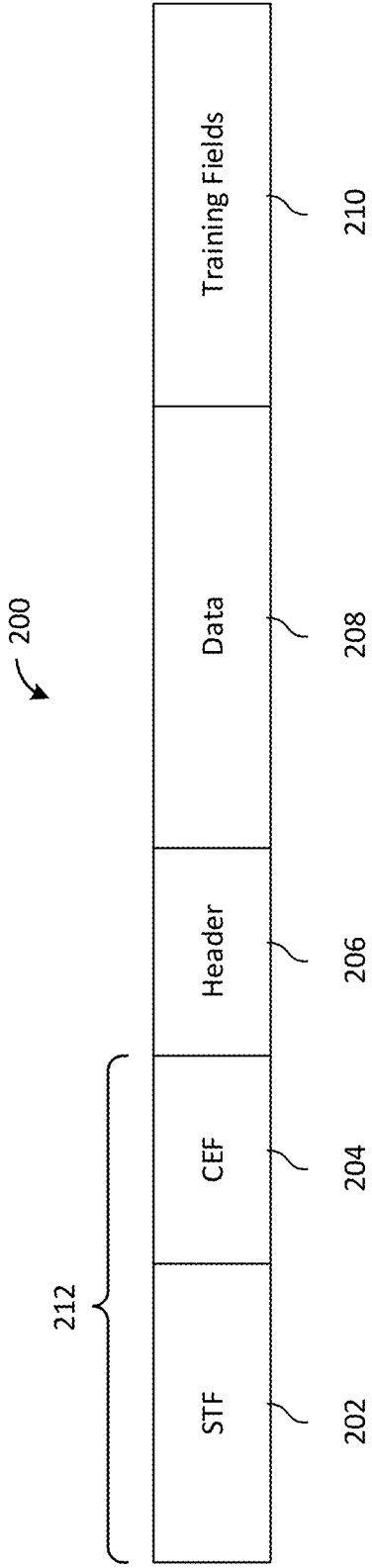


Fig. 2

Ga ₁₂₈															
+1	+1	-1	-1	-1	-1	-1	-1	-1	+1	-1	+1	+1	-1	-1	+1
+1	+1	-1	-1	+1	+1	+1	+1	-1	+1	-1	+1	-1	+1	+1	-1
-1	-1	+1	+1	+1	+1	+1	+1	+1	-1	+1	-1	-1	+1	+1	-1
+1	+1	-1	-1	+1	+1	+1	+1	-1	+1	-1	+1	-1	+1	+1	-1
+1	+1	-1	-1	-1	-1	-1	-1	-1	+1	-1	+1	+1	-1	-1	+1
+1	+1	-1	-1	+1	+1	+1	+1	-1	+1	-1	+1	-1	+1	+1	-1
+1	+1	-1	-1	-1	-1	-1	-1	-1	+1	-1	+1	+1	-1	-1	+1
-1	-1	+1	+1	-1	-1	-1	-1	+1	-1	+1	-1	+1	-1	-1	+1

Fig. 3A

Gb ₁₂₈															
-1	-1	+1	+1	+1	+1	+1	+1	+1	-1	+1	-1	-1	+1	+1	-1
-1	-1	+1	+1	-1	-1	-1	-1	+1	-1	+1	-1	+1	-1	-1	+1
+1	+1	-1	-1	-1	-1	-1	-1	-1	+1	-1	+1	+1	-1	-1	+1
-1	-1	+1	+1	-1	-1	-1	-1	+1	-1	+1	-1	+1	-1	-1	+1
+1	+1	-1	-1	-1	-1	-1	-1	-1	+1	-1	+1	+1	-1	-1	+1
+1	+1	-1	-1	+1	+1	+1	+1	-1	+1	-1	+1	-1	+1	+1	-1
+1	+1	-1	-1	-1	-1	-1	-1	-1	+1	-1	+1	+1	-1	-1	+1
-1	-1	+1	+1	-1	-1	-1	-1	+1	-1	+1	-1	+1	-1	-1	+1

Fig. 3B

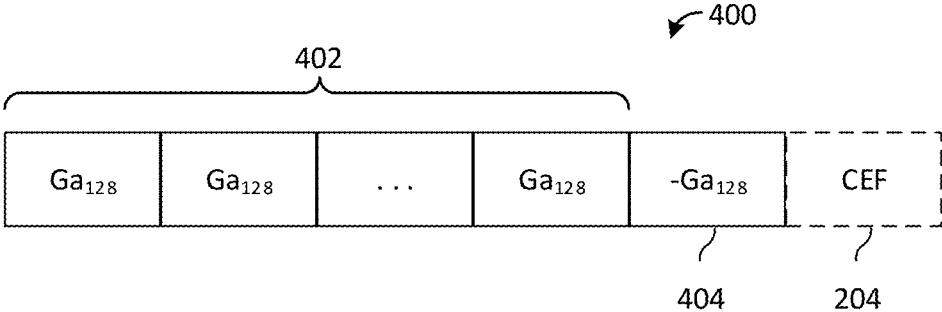


Fig. 4A

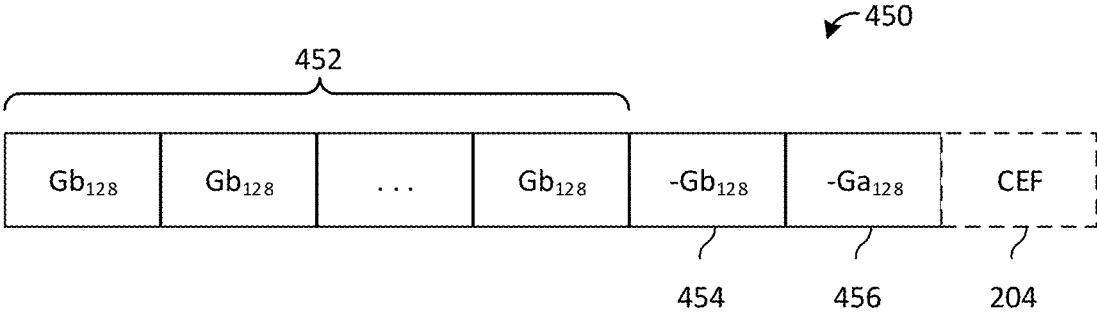


Fig. 4B

802.11ad non-control STF & Ga₁₂₈

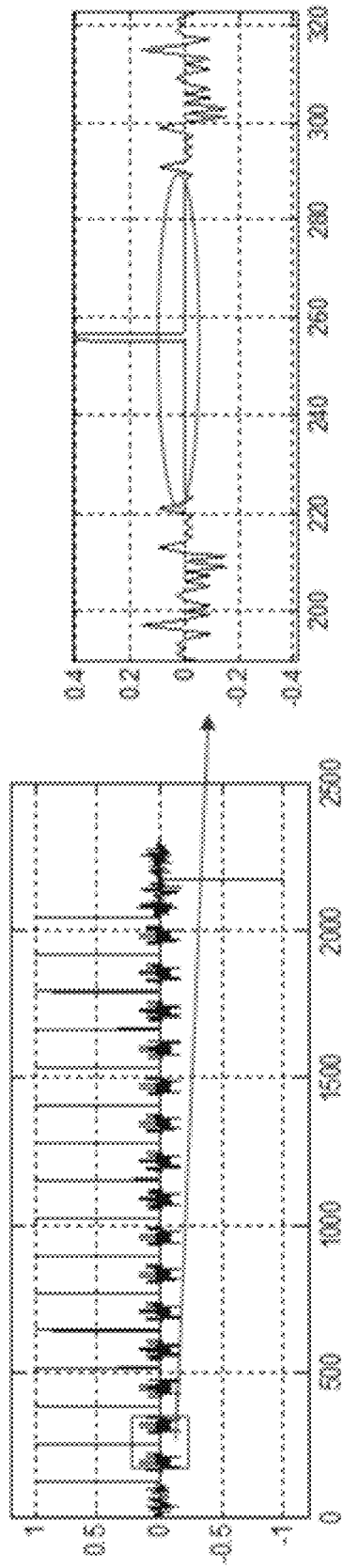


Fig. 5A

802.11ad non-control STF & Gb₁₂₈

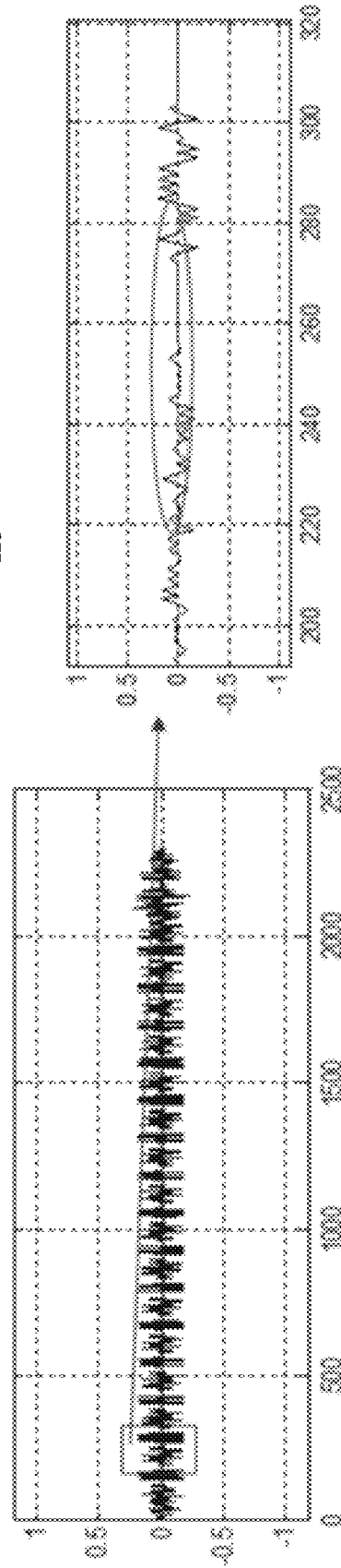


Fig. 5B

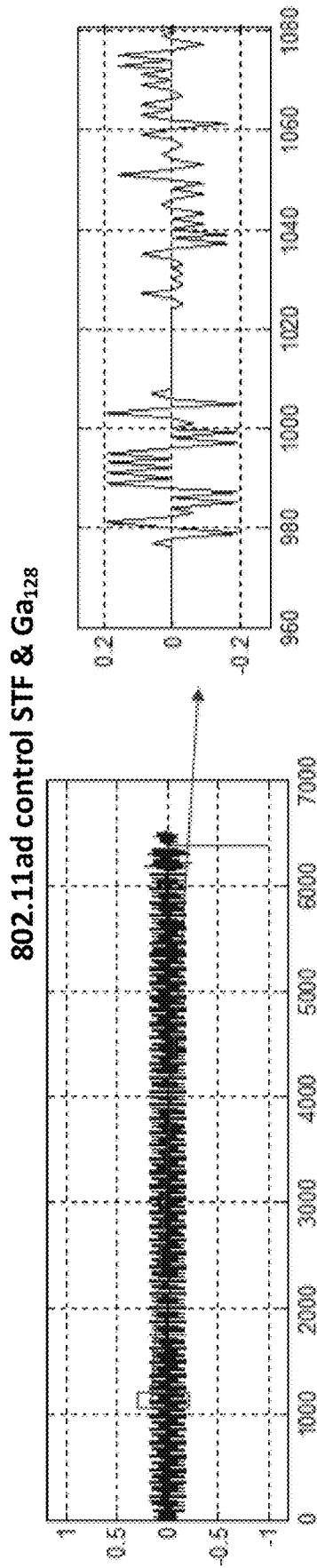


Fig. 5C

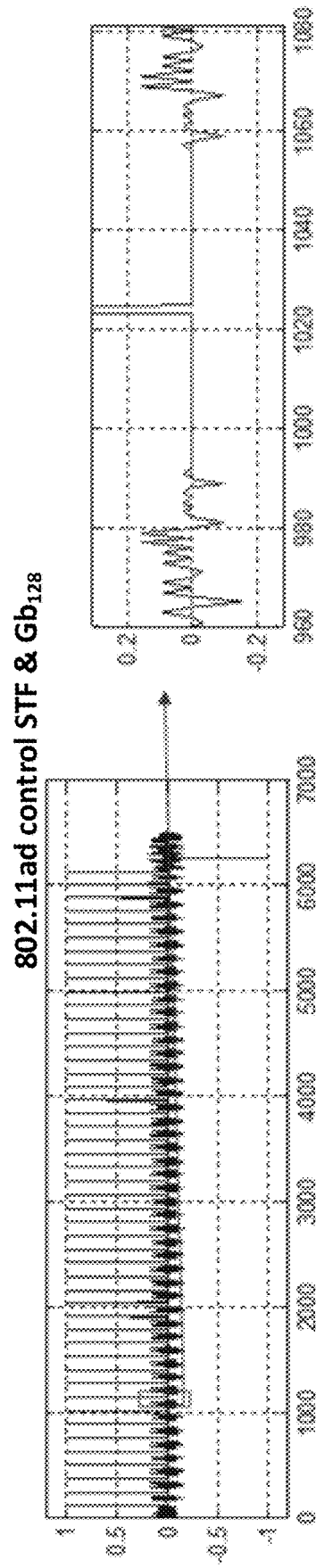


Fig. 5D

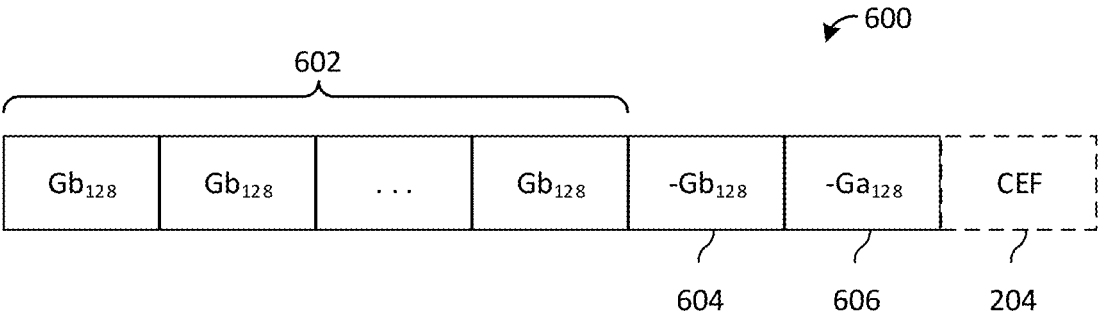


Fig. 6A

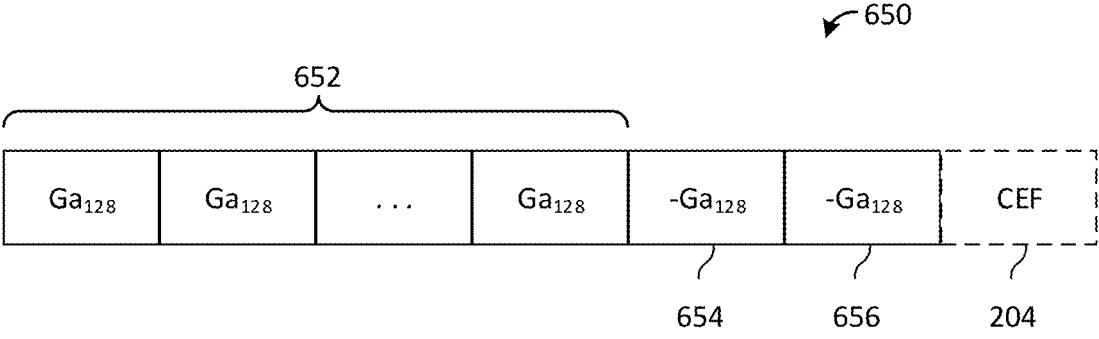


Fig. 6B

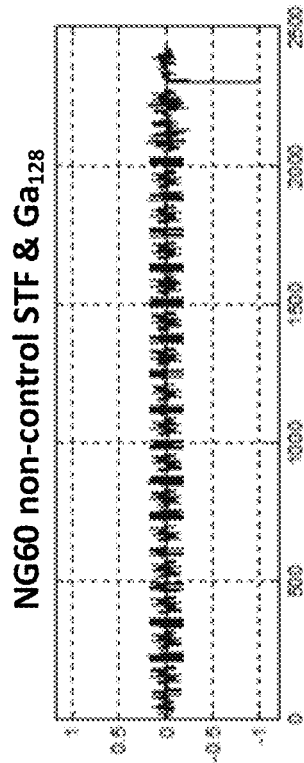


Fig. 7A

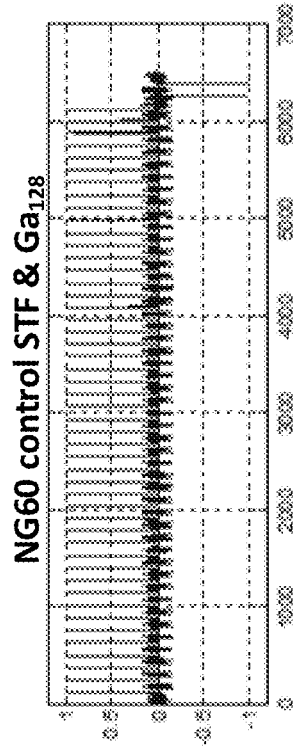


Fig. 7C

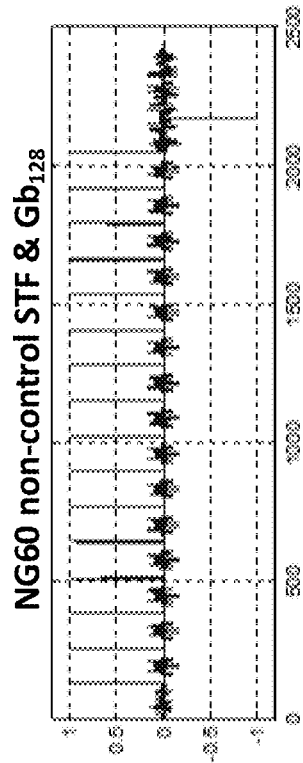


Fig. 7B

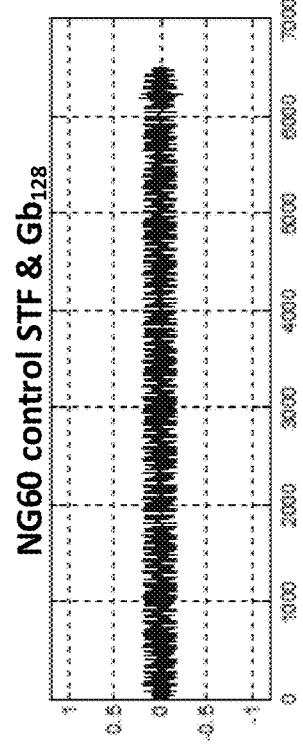


Fig. 7D

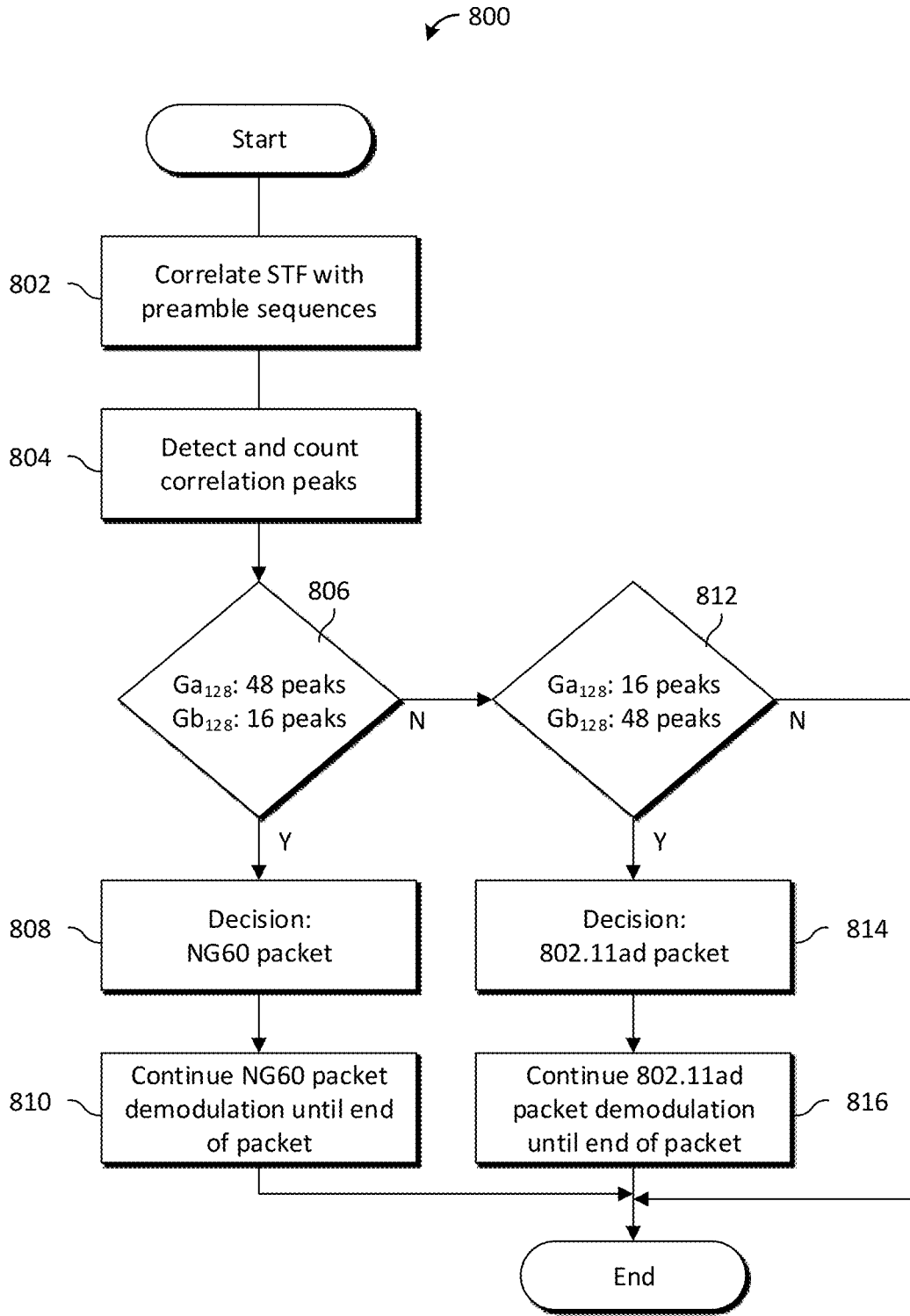


Fig. 8A

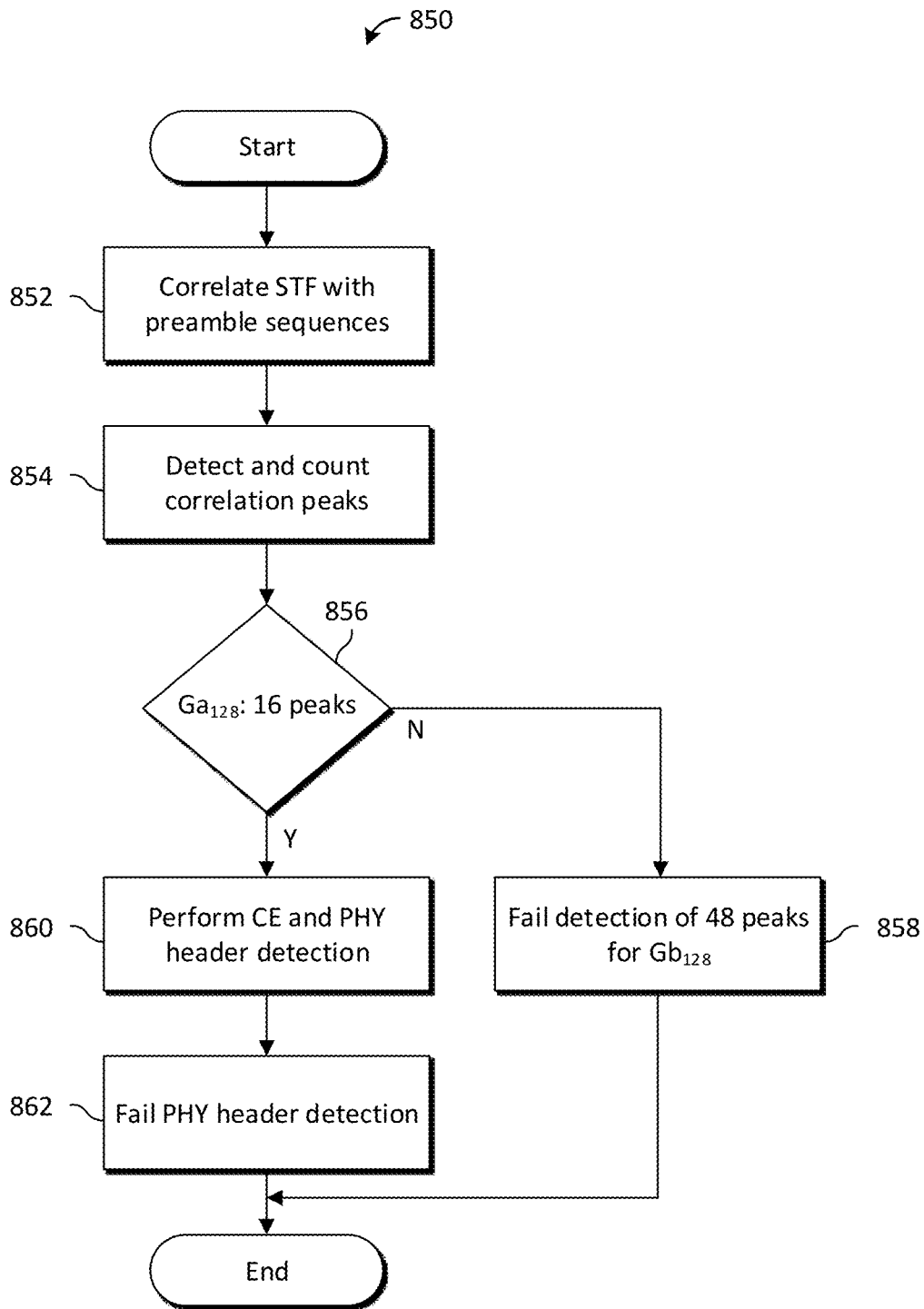


Fig. 8B

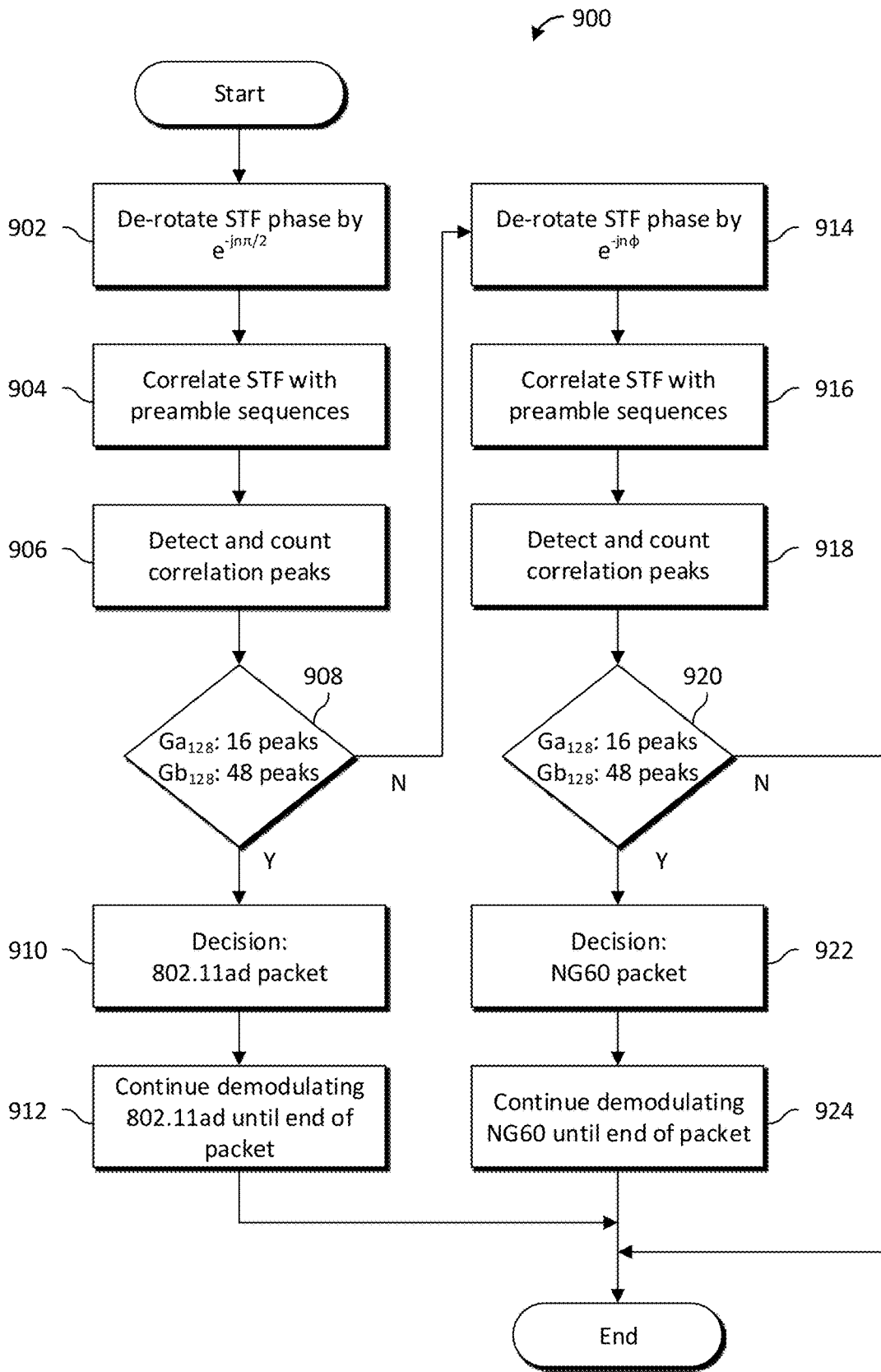


Fig. 9A

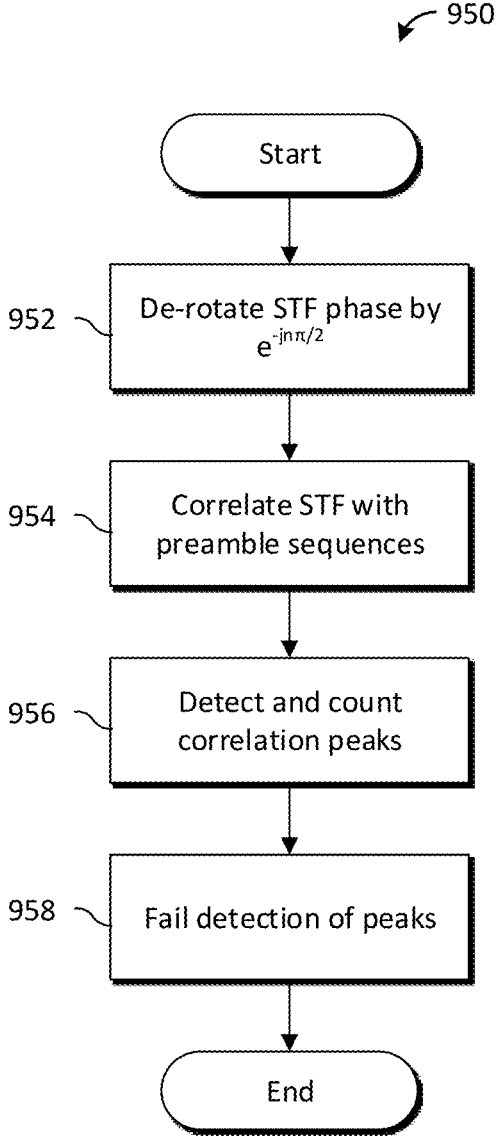


Fig. 9B

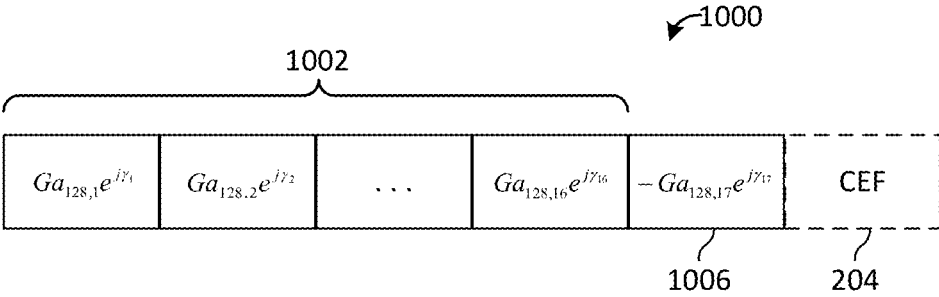


Fig. 10A

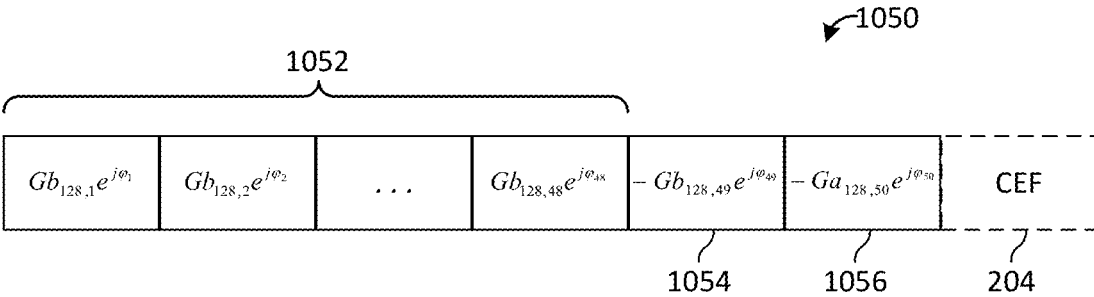


Fig. 10B

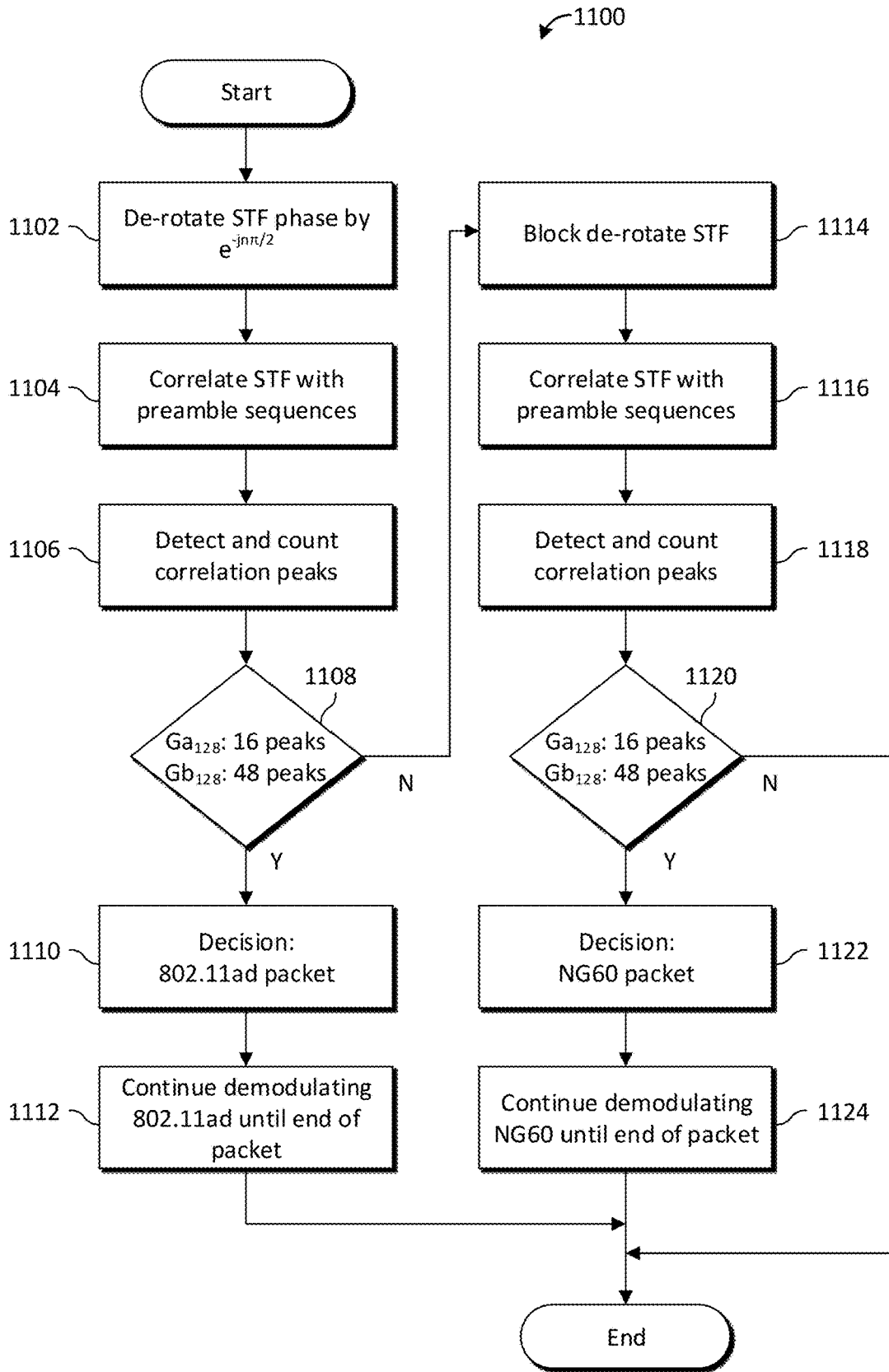


Fig. 11A

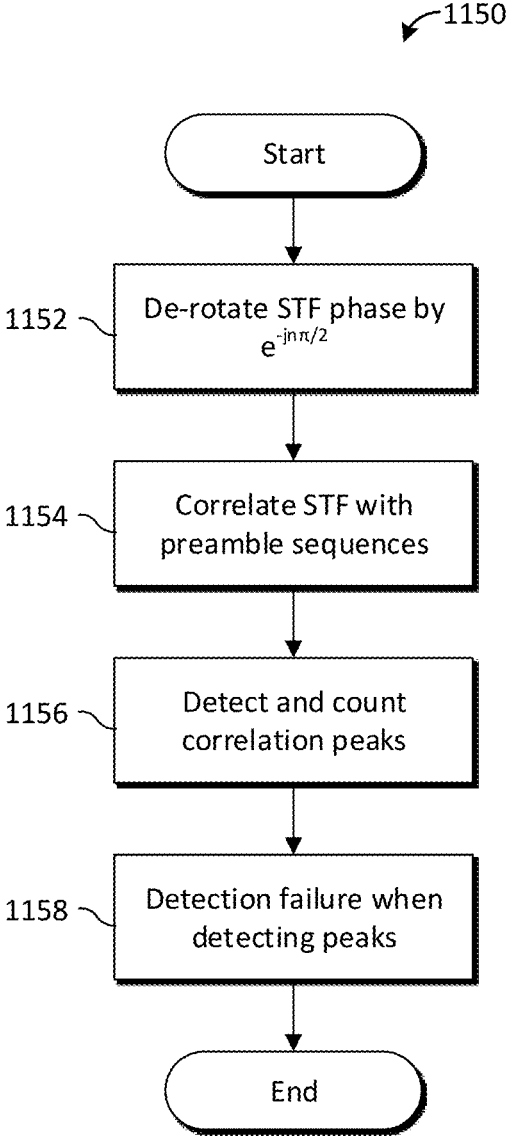


Fig. 11B

Ga_{128, new, 1}															
+1	-1	-1	+1	-1	+1	-1	+1	-1	-1	-1	-1	+1	+1	-1	-1
+1	-1	-1	+1	+1	-1	+1	-1	-1	-1	-1	-1	-1	-1	+1	+1
-1	+1	+1	-1	+1	-1	+1	-1	+1	+1	+1	+1	-1	-1	+1	+1
+1	-1	-1	+1	+1	-1	+1	-1	-1	-1	-1	-1	-1	-1	+1	+1
+1	-1	-1	+1	-1	+1	-1	+1	-1	-1	-1	-1	+1	+1	-1	-1
+1	-1	-1	+1	+1	-1	+1	-1	-1	-1	-1	-1	-1	-1	+1	+1
+1	-1	-1	+1	-1	+1	-1	+1	-1	-1	-1	-1	+1	+1	-1	-1
-1	+1	+1	-1	-1	+1	-1	+1	+1	+1	+1	+1	+1	+1	-1	-1

Fig. 12A

Gb_{128, new, 1}															
-1	+1	+1	-1	+1	-1	+1	-1	+1	+1	+1	+1	-1	-1	+1	+1
-1	+1	+1	-1	-1	+1	-1	+1	+1	+1	+1	+1	+1	+1	-1	-1
+1	-1	-1	+1	-1	+1	-1	+1	-1	-1	-1	-1	+1	+1	-1	-1
-1	+1	+1	-1	-1	+1	-1	+1	+1	+1	+1	+1	+1	+1	-1	-1
+1	-1	-1	+1	-1	+1	-1	+1	-1	-1	-1	-1	+1	+1	-1	-1
+1	-1	-1	+1	+1	-1	+1	-1	-1	-1	-1	-1	-1	-1	+1	+1
+1	-1	-1	+1	-1	+1	-1	+1	-1	-1	-1	-1	+1	+1	-1	-1
-1	+1	+1	-1	-1	+1	-1	+1	+1	+1	+1	+1	+1	+1	-1	-1

Fig. 12B

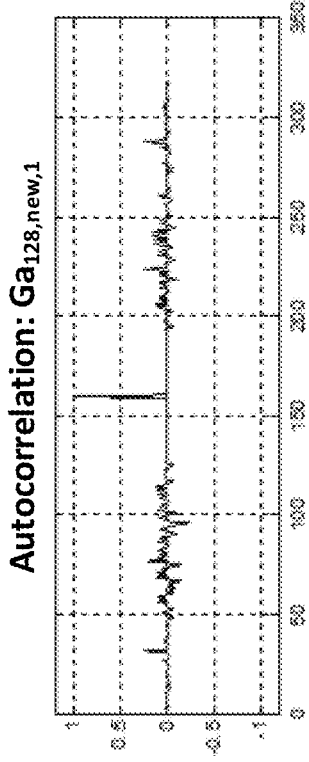


Fig. 13A

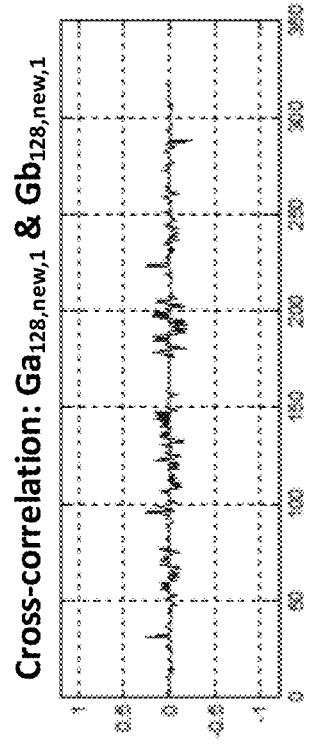


Fig. 13C

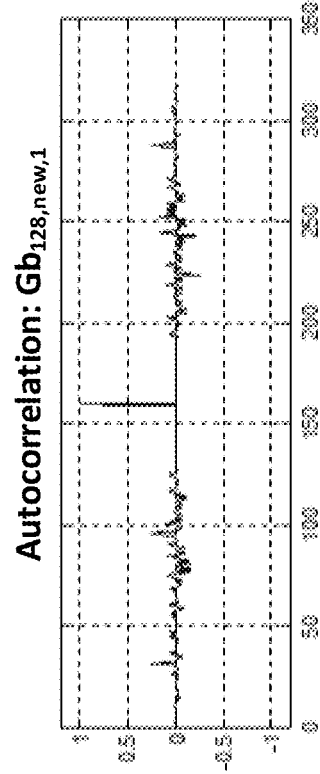


Fig. 13B

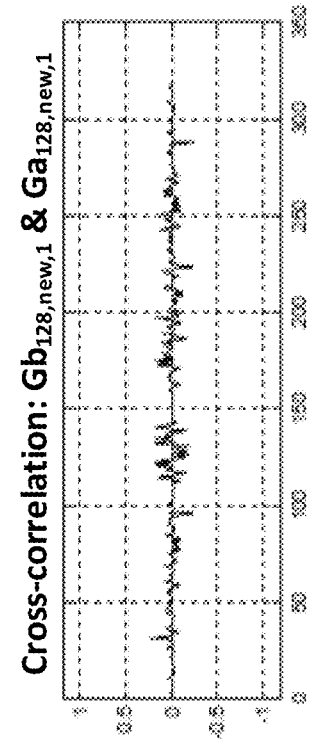


Fig. 13D

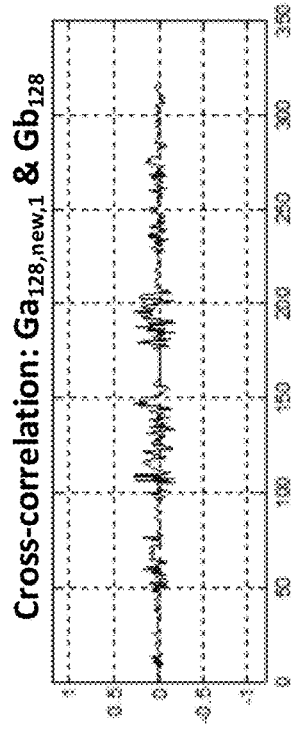


Fig. 13G

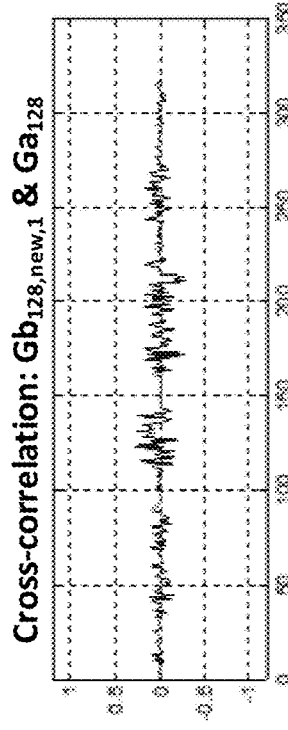


Fig. 13H

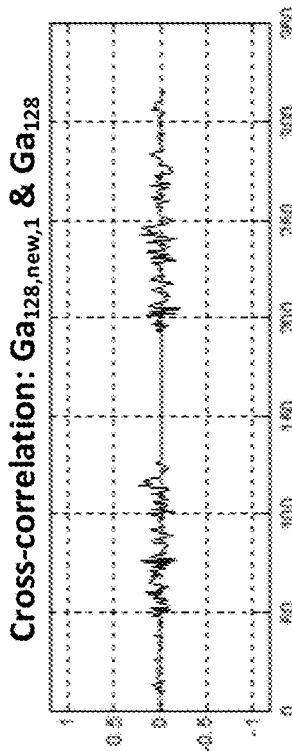


Fig. 13E

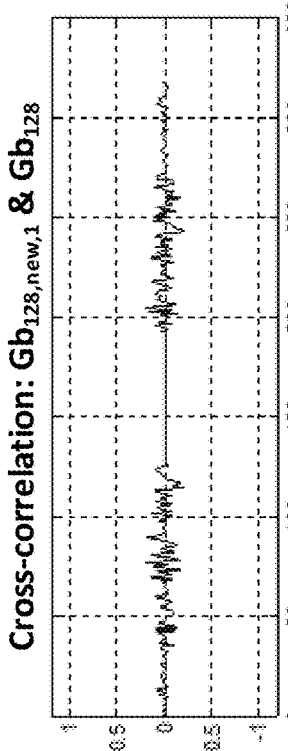


Fig. 13F

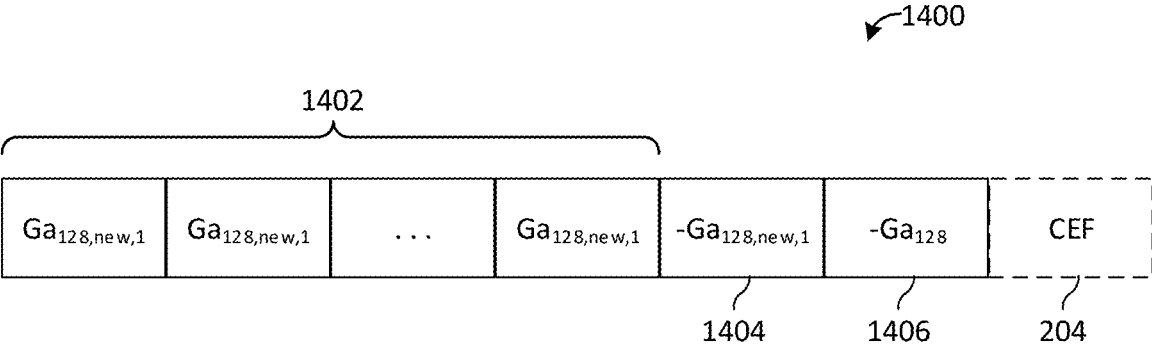


Fig. 14A

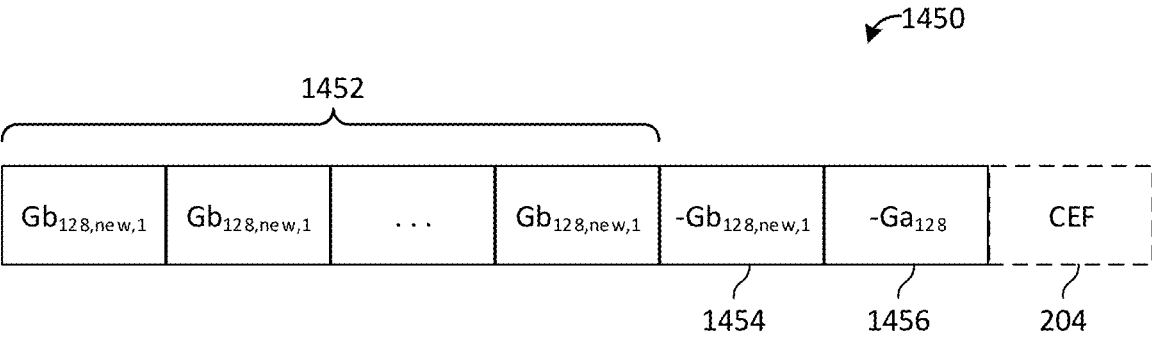


Fig. 14B

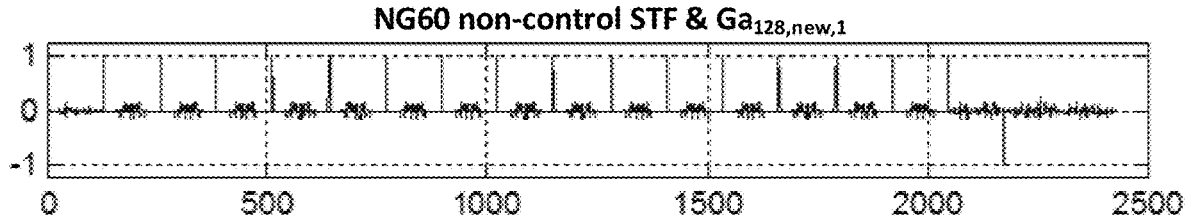


Fig. 15A

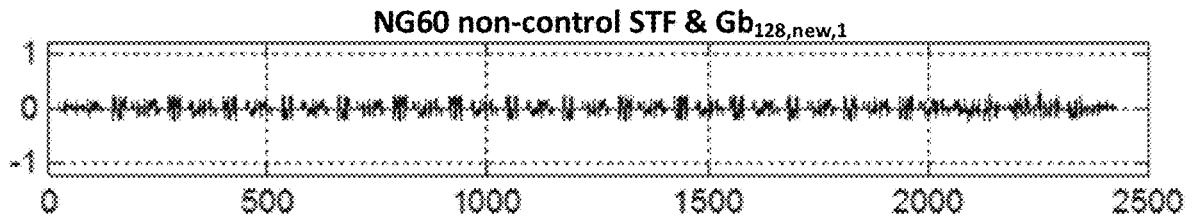


Fig. 15B

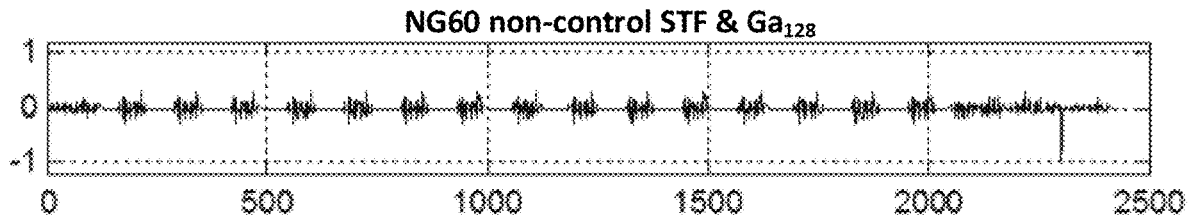


Fig. 15C

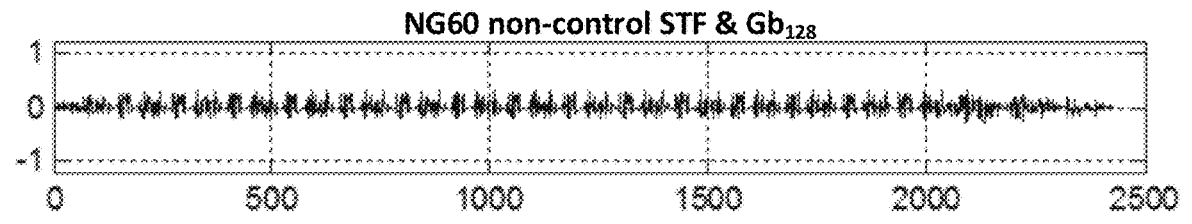


Fig. 15D

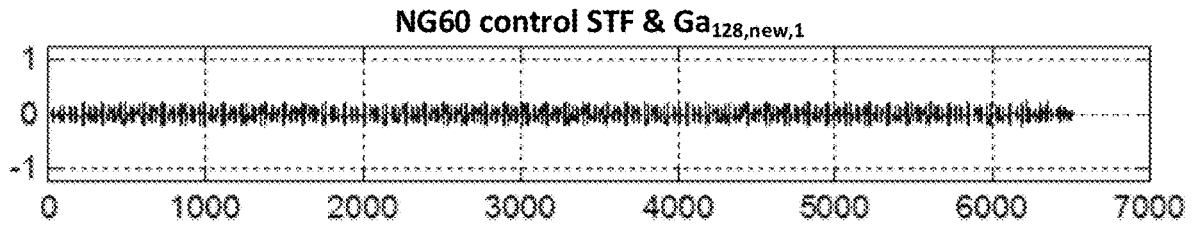


Fig. 15E

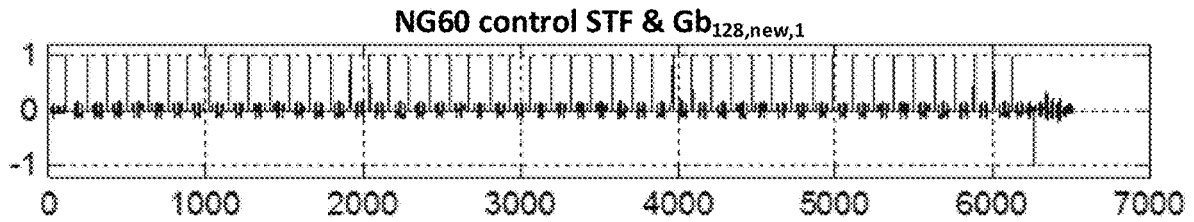


Fig. 15F

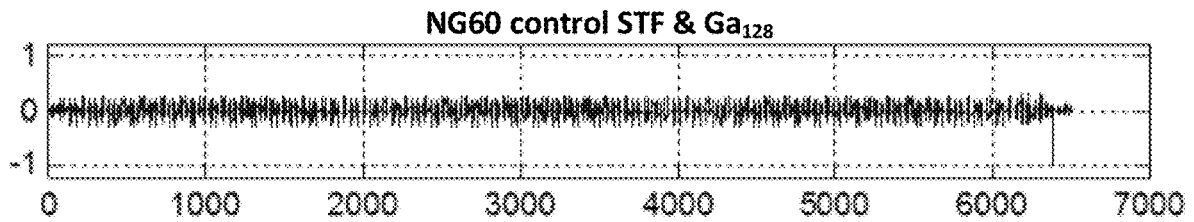


Fig. 15G

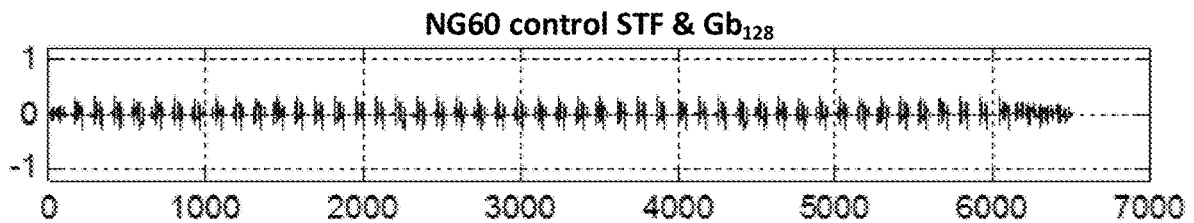


Fig. 15H

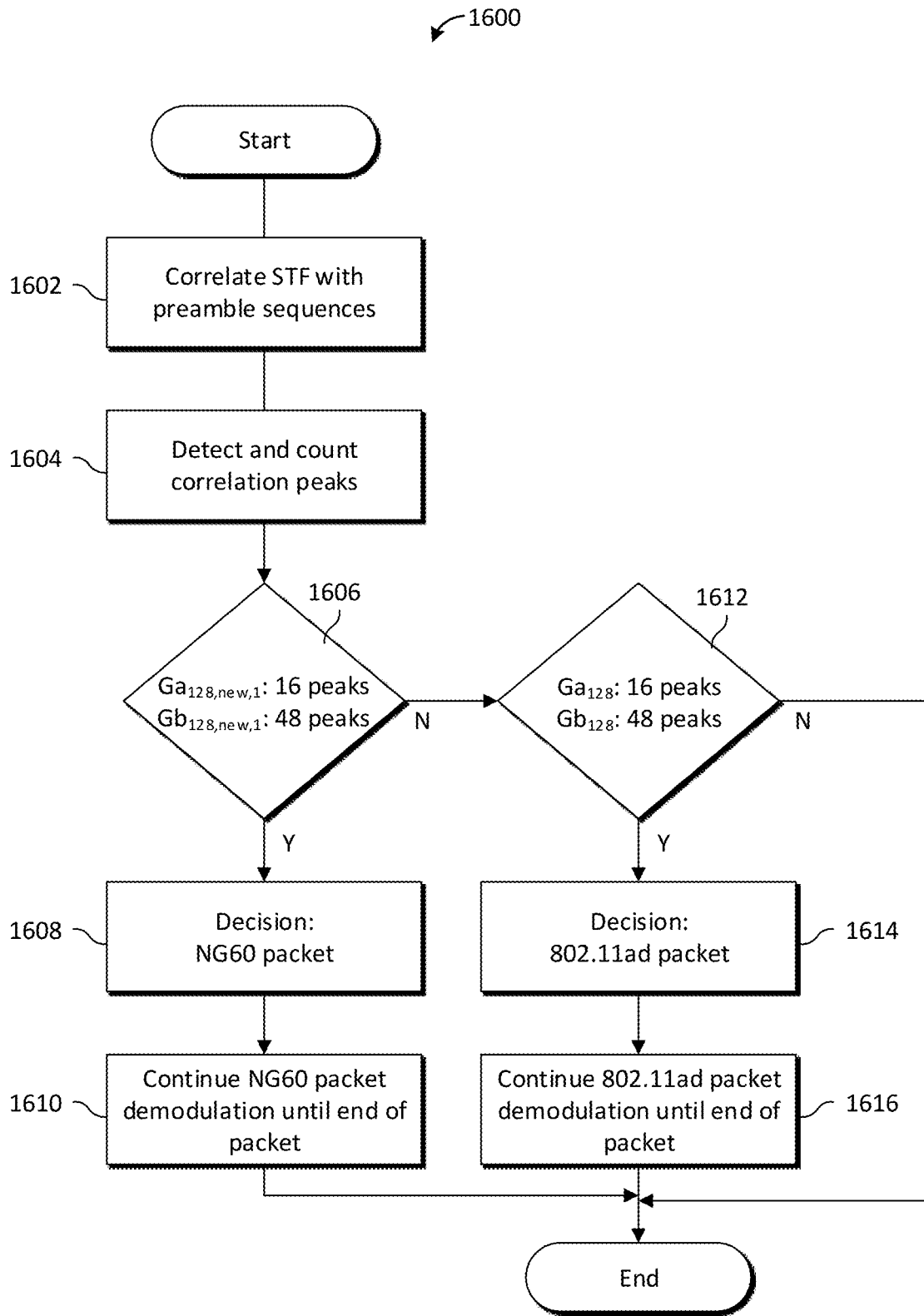


Fig. 16A

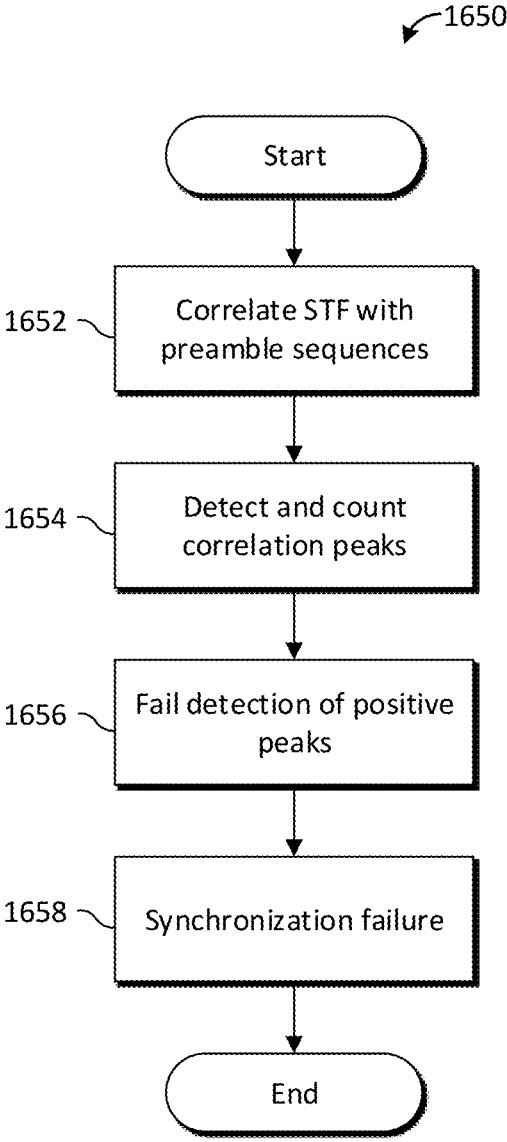


Fig. 16B

Ga_{128, new, 2}															
-1	+1	-1	-1	+1	-1	-1	-1	+1	-1	+1	+1	+1	-1	-1	-1
-1	+1	-1	-1	+1	-1	-1	-1	-1	+1	-1	-1	-1	+1	+1	+1
+1	-1	+1	+1	-1	+1	+1	+1	-1	+1	-1	-1	-1	+1	+1	+1
+1	-1	+1	+1	-1	+1	+1	+1	+1	-1	+1	+1	+1	-1	-1	-1
+1	-1	+1	+1	-1	+1	+1	+1	-1	+1	-1	-1	-1	+1	+1	+1
-1	+1	-1	-1	+1	-1	-1	-1	-1	+1	-1	-1	-1	+1	+1	+1
+1	-1	+1	+1	-1	+1	+1	+1	-1	+1	-1	-1	-1	+1	+1	+1
-1	+1	-1	-1	+1	-1	-1	-1	-1	+1	-1	-1	-1	+1	+1	+1

Fig. 17A

Gb_{128, new, 2}															
+1	+1	+1	-1	-1	-1	+1	-1	-1	-1	-1	+1	-1	-1	+1	-1
+1	+1	+1	-1	-1	-1	+1	-1	+1	+1	+1	-1	+1	+1	-1	+1
-1	-1	-1	+1	+1	+1	-1	+1	+1	+1	+1	-1	+1	+1	-1	+1
-1	-1	-1	+1	+1	+1	-1	+1	-1	-1	-1	+1	-1	-1	+1	-1
-1	-1	-1	+1	+1	+1	-1	+1	+1	+1	+1	-1	+1	+1	-1	+1
+1	+1	+1	-1	-1	-1	+1	-1	+1	+1	+1	-1	+1	+1	-1	+1
-1	-1	-1	+1	+1	+1	-1	+1	+1	+1	+1	-1	+1	+1	-1	+1
+1	+1	+1	-1	-1	-1	+1	-1	+1	+1	+1	-1	+1	+1	-1	+1

Fig. 17B

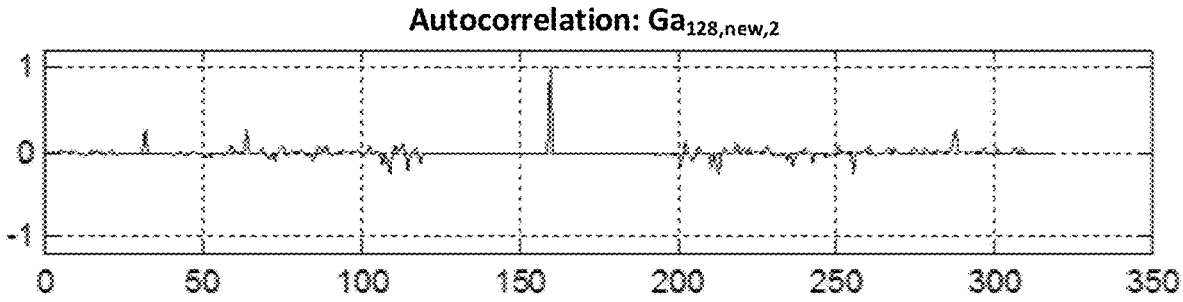


Fig. 18A

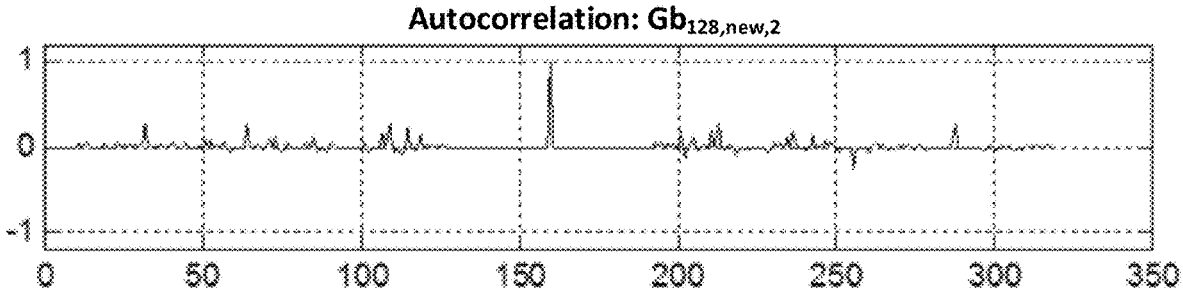


Fig. 18B

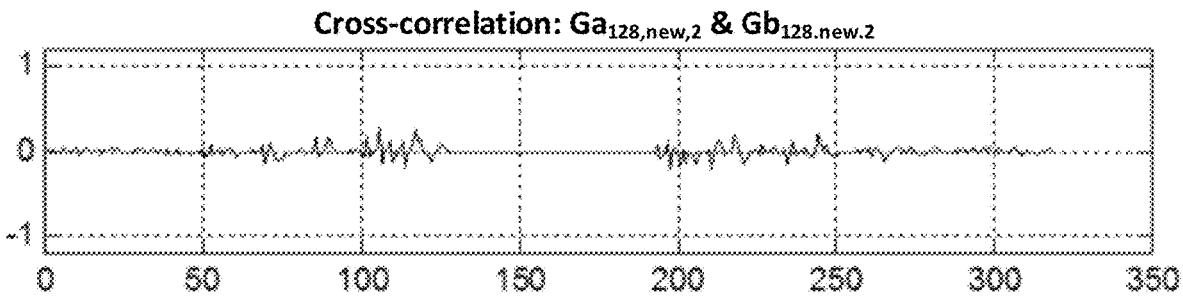


Fig. 18C

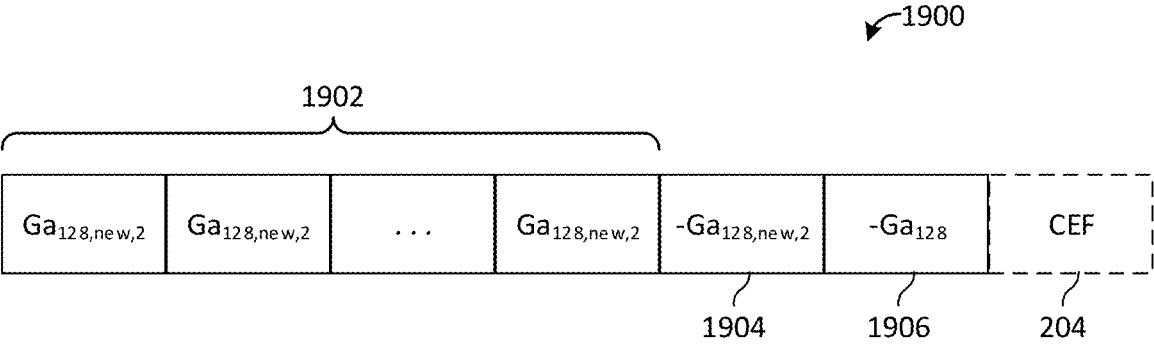


Fig. 19A

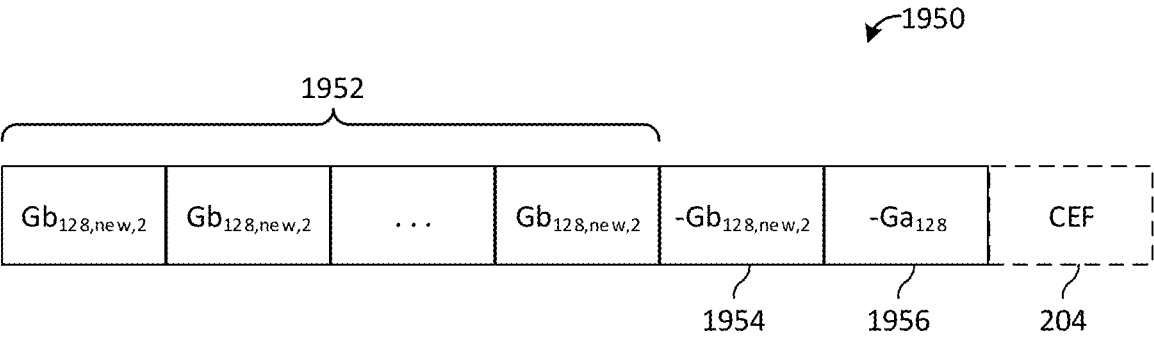


Fig. 19B

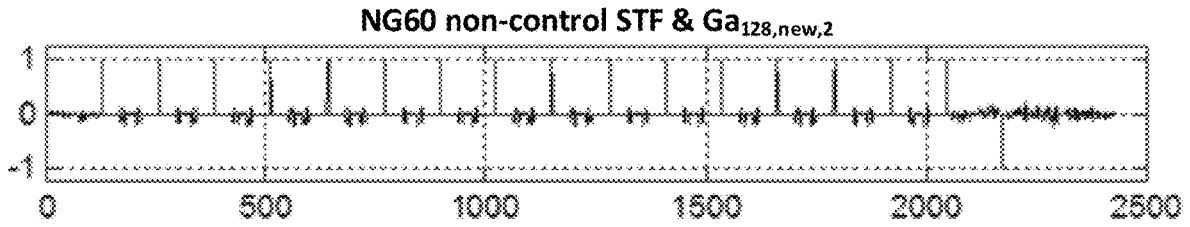


Fig. 20A

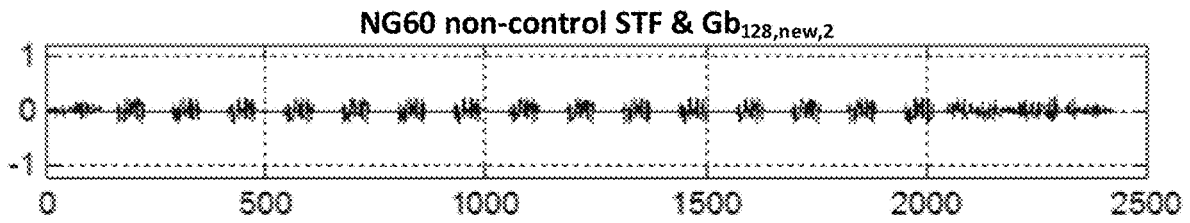


Fig. 20B

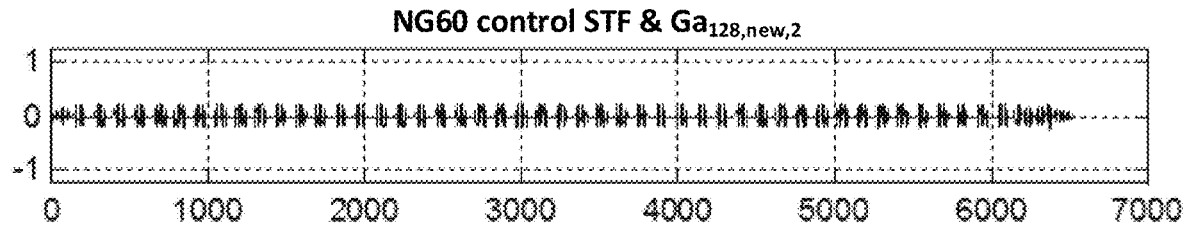


Fig. 20C

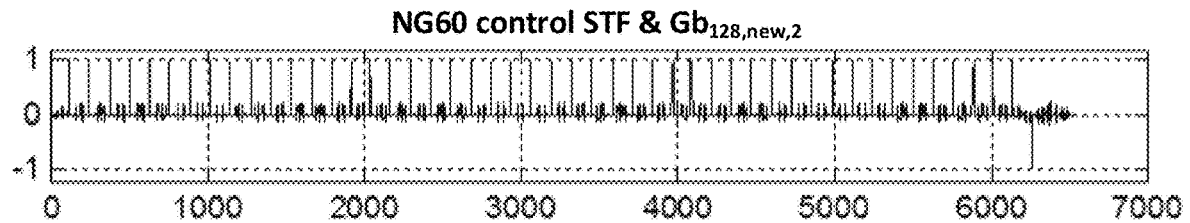


Fig. 20D

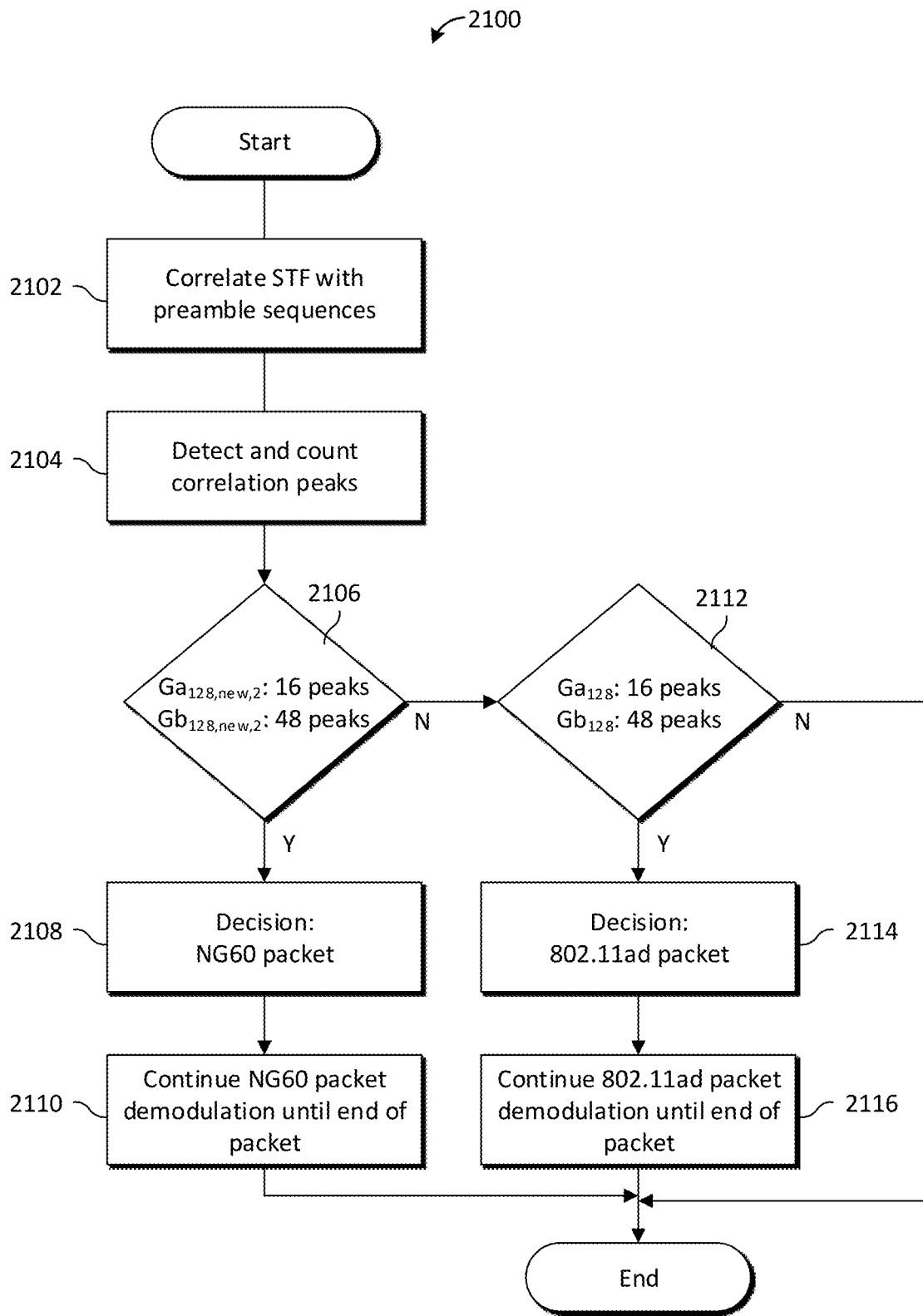


Fig. 21A

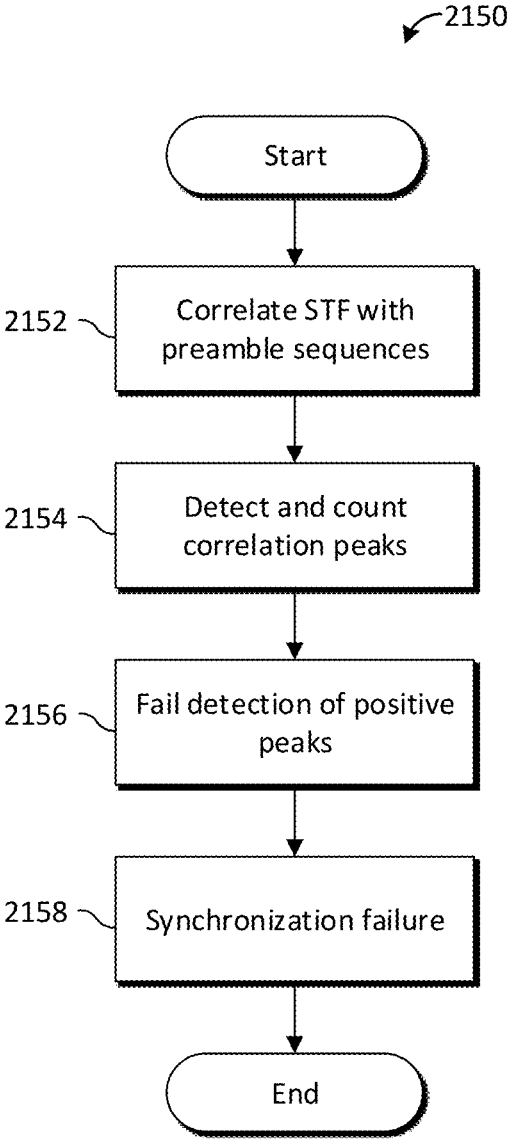


Fig. 21B

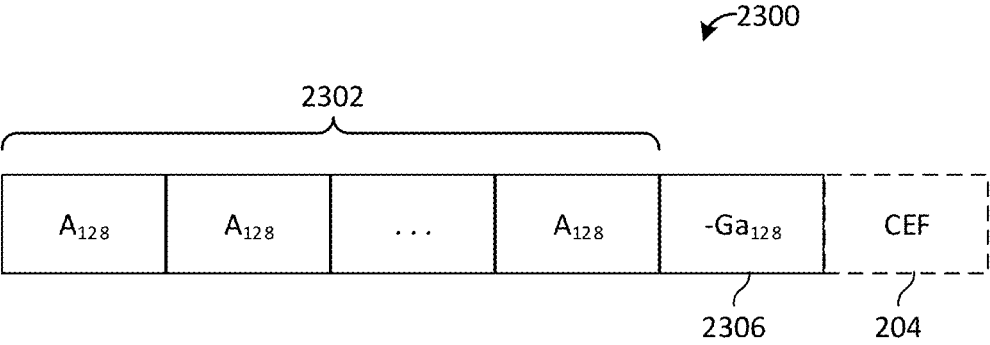


Fig. 23A

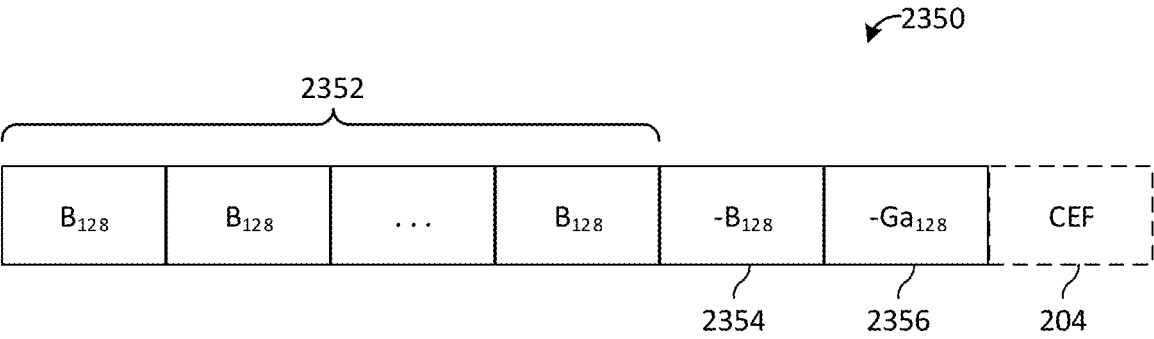


Fig. 23B

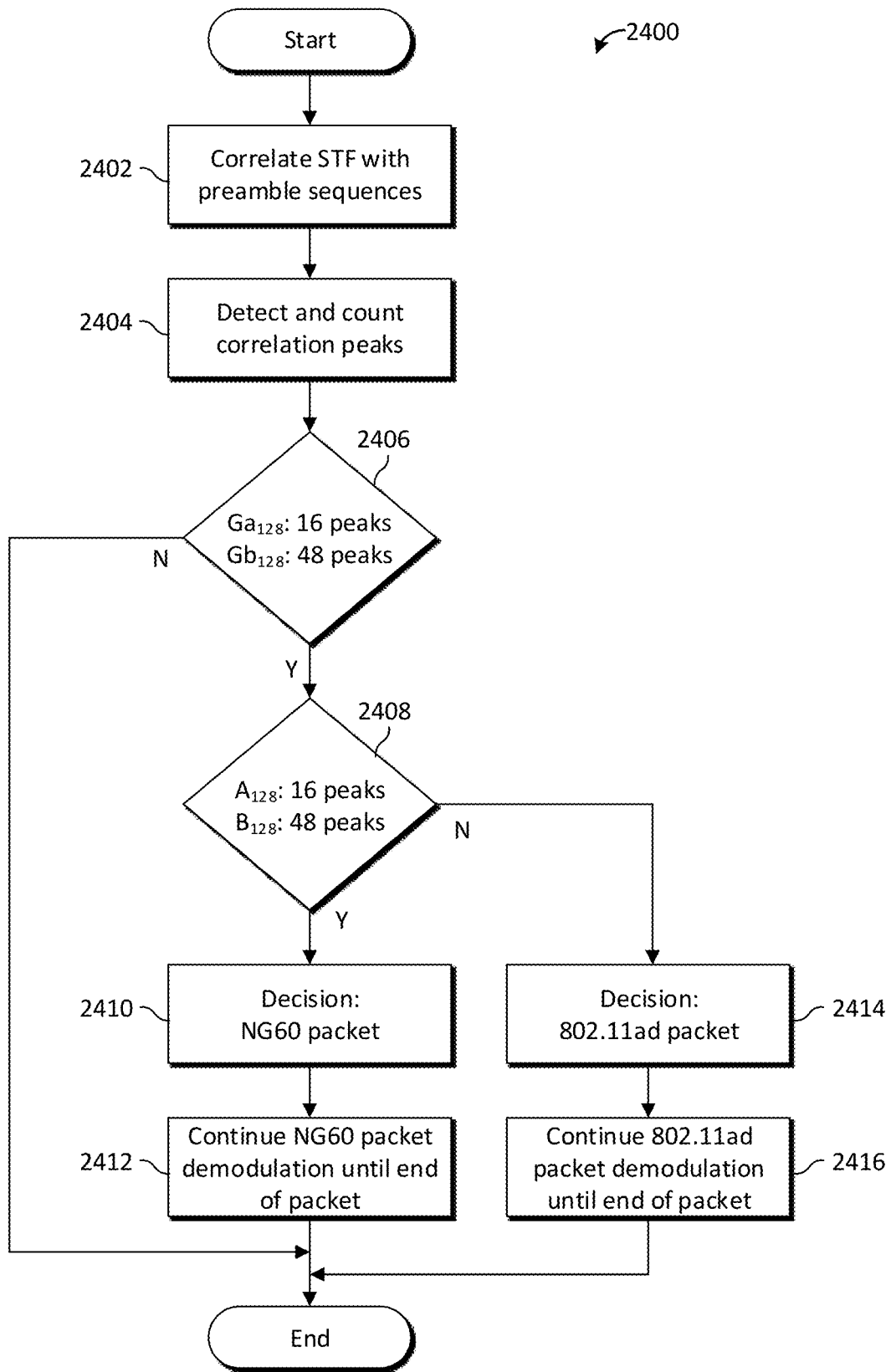


Fig. 24A

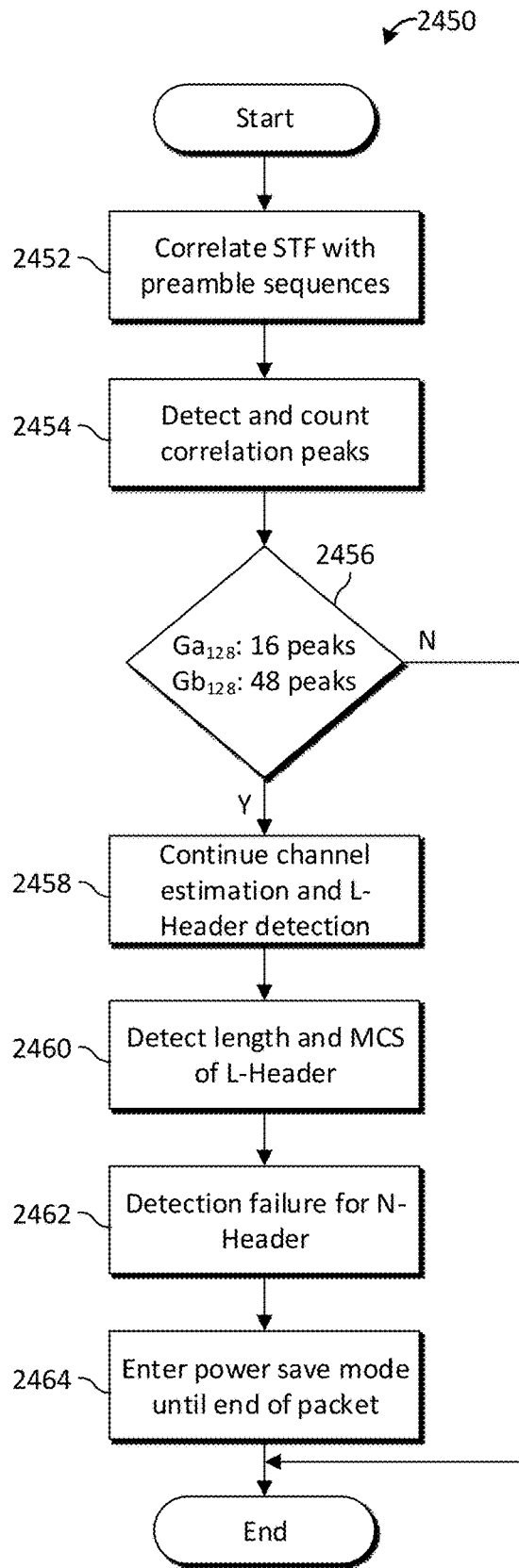


Fig. 24B

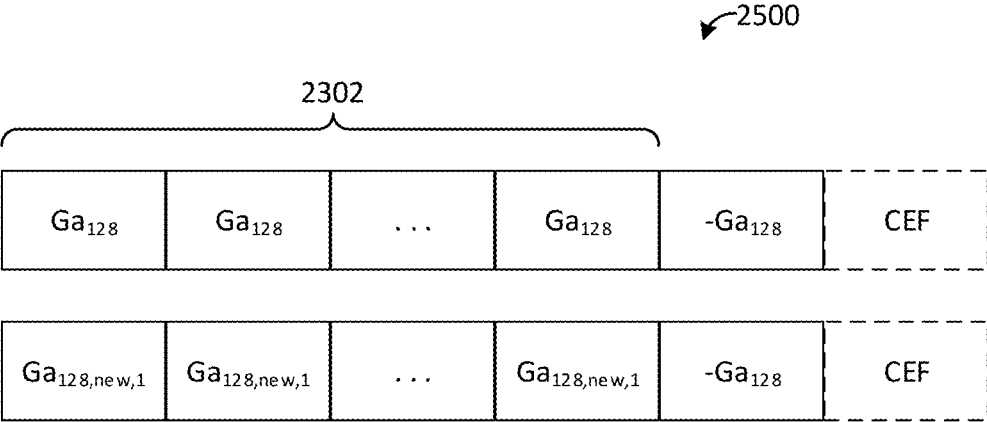


Fig. 25A

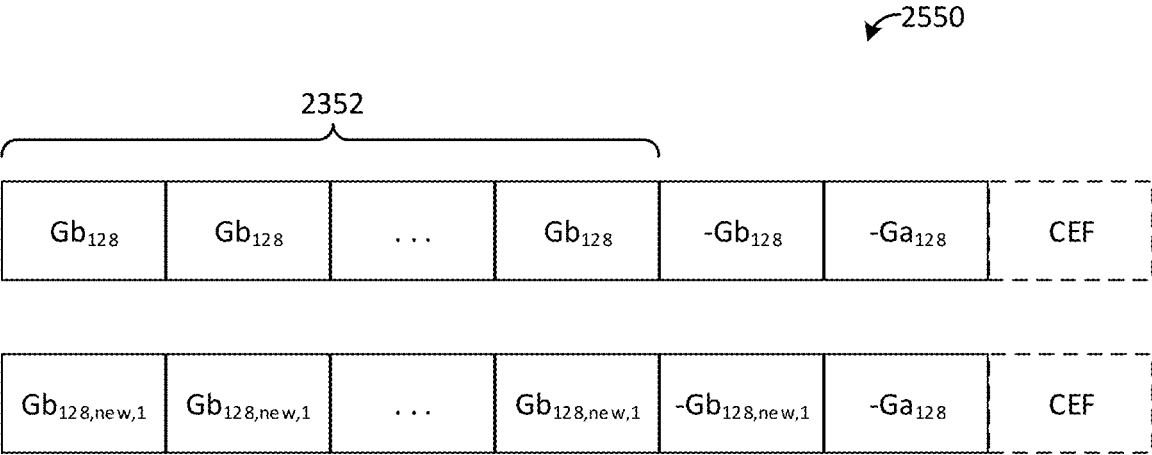


Fig. 25B

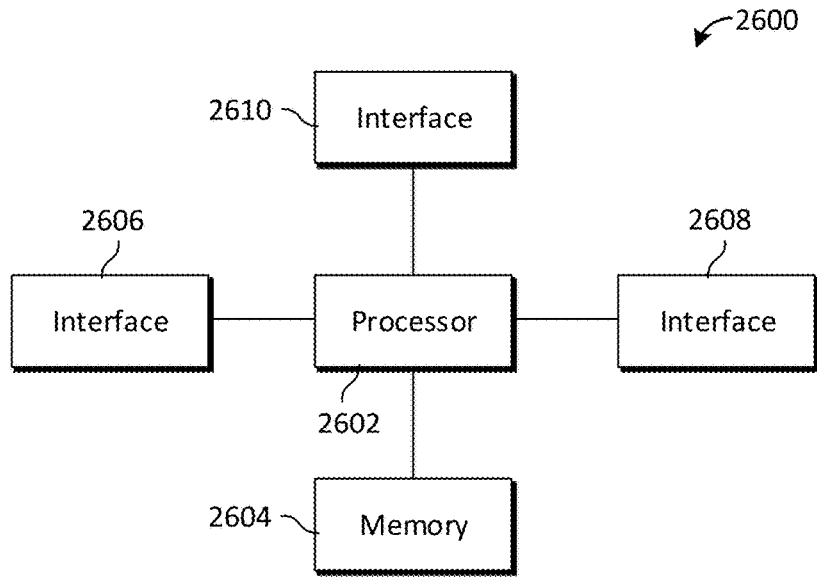


Fig. 26

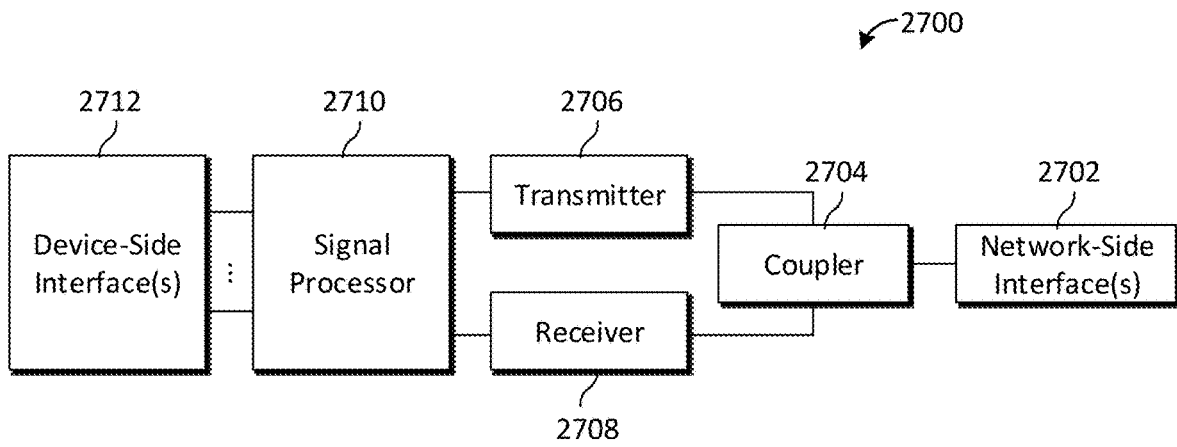


Fig. 27

METHOD FOR AUTO-DETECTION OF WLAN PACKETS USING STF

[0001] This application is a continuation of U.S. application Ser. No. 17/028,592, filed on Sep. 22, 2020, and entitled “Method for Auto-Detection of WLAN Packets using STF,” which is a divisional of U.S. application Ser. No. 15/043,089, filed on Feb. 12, 2016, now U.S. Pat. No. 10,813,083 issued on Oct. 20, 2020, and entitled “System and Method for Auto-Detection of WLAN Packets using STF,” which claims priority to U.S. Provisional Application No. 62/115,445, filed on Feb. 12, 2015, and entitled “Method and Apparatus for Auto-Detection of WLAN Packets,” and U.S. Provisional Application No. 62/219,794, filed on Sep. 17, 2015, and entitled “System and Method for Auto-Detection of 60 GHz WLAN Packets in STF,” all of which applications are hereby incorporated by reference herein in their entireties.

TECHNICAL FIELD

[0002] The present invention relates generally to wireless networks, and in particular embodiments, to techniques and mechanisms for auto-detection of wireless local area network (WLAN) packets.

BACKGROUND

[0003] With the increasing demands of high-definition (HD) displays and other applications, and with the widespread usage of smart phones and tablets, next generation WLANs capable of transmission at higher data rates are needed. IEEE 802.11ad is a WLAN technique that operates in the globally unlicensed 60 GHz band, e.g., 57-66 GHz. Next generation 60 GHz WLANs (NG60), capable of even higher performance than 802.11ad, have been proposed. Although such networks may be capable of higher performance, there is also a need for them to be compatible with current 802.11ad devices.

SUMMARY OF THE INVENTION

[0004] Technical advantages are generally achieved by embodiments of this disclosure, which describe auto-detection of WLAN packets.

[0005] In accordance with an embodiment, a method is provided. The method includes receiving a wireless packet in a 60GHz frequency band comprising a short training field (STF), determining a first quantity of cross-correlation peaks between the STF and a first preamble component sequence, and a second quantity of cross-correlation peaks between the STF and a second preamble component sequence, the first quantity being a first threshold or the second quantity being a second threshold for a first packet type, the first quantity being the second threshold or the second quantity being the first threshold for a second packet type, and determining that the wireless packet is the second packet type when the first quantity of cross-correlation peaks equals the second threshold or the second quantity of cross-correlation peaks equals the first threshold.

[0006] In some embodiments, the first threshold is forty-eight and the second threshold is sixteen. In some embodiments, the first packet type is an Institute of Electrical and Electronics Engineers (IEEE) 802.11ad packet type, and the second packet type is an IEEE 802.11ay packet type. In

some embodiments, the first preamble component sequence is $G_{a_{128}}$ and the second preamble component sequence is $G_{b_{128}}$.

[0007] In accordance with an embodiment, a method is provided. The method includes transmitting in a 60 GHz frequency band a wireless packet comprising a short training field (STF), a header, a payload, and a training field, the STF producing a first quantity of cross-correlation peaks when correlated with a first preamble component sequence, the STF producing a second quantity of cross-correlation peaks when correlated with a second preamble component sequence, the first quantity being a first threshold or the second quantity being a second threshold for a first packet type, the first quantity being the second threshold or the second quantity being the first threshold for a second packet type.

[0008] In some embodiments, the first threshold is forty-eight and the second threshold is sixteen. In some embodiments, the first packet type is an Institute of Electrical and Electronics Engineers (IEEE) 802.11ad packet type, and the second packet type is an IEEE 802.11ay packet type. In some embodiments, the first preamble component sequence is $G_{b_{128}}$ and the second preamble component sequence is $G_{a_{128}}$.

[0009] In accordance with an embodiment, a method is provided. The method includes receiving a wireless packet in a 60 GHz frequency band comprising a short training field (STF), determining a first quantity of cross-correlation peaks between the STF and a first preamble component sequence, and a second quantity of cross-correlation peaks between the STF and a second preamble component sequence, the first quantity being a first threshold or the second quantity being a second threshold for a first packet type, removing a phase shift from the STF to produce a phase shifted STF, determining a third quantity of cross-correlation peaks between the phase shifted STF and the first preamble component sequence, and a fourth quantity of cross-correlation peaks between the phase shifted STF and the second preamble component sequence, and determining that the wireless packet is a second packet type when the third quantity equals the first threshold or the fourth quantity equals the second threshold.

[0010] In some embodiments, the first packet type is an Institute of Electrical and Electronics Engineers (IEEE) 802.11ad packet type, and the second packet type is an IEEE 802.11ay packet type. In some embodiments, removing the phase shift from the STF comprises applying a linear phase rotation other than

$$e^{j\pi \frac{n}{2}}$$

to the STF. In some embodiments, removing the phase shift from the STF comprises applying a block rotation to the STF.

[0011] In accordance with an embodiment, a method is provided. The method includes adding a phase shift to a short training field (STF) to produce a phase shifted STF, and transmitting in a 60 GHz frequency band a wireless packet comprising the phase shifted STF, a header, a payload, and a training field, the phase shifted STF producing a first quantity of cross-correlation peaks when correlated with a first preamble component sequence and a second quantity

of cross-correlation peaks when correlated with a second preamble component sequence, the phase shifted STF producing a third quantity of cross-correlation peaks when correlated with the first preamble component sequence and a fourth quantity of cross-correlation peaks when correlated with the second preamble component sequence after the phase shift is removed from the phase shifted STF, the first quantity being a first threshold or the second quantity being a second threshold for a first packet type, the third quantity being the first threshold or the fourth quantity being the second threshold for a second packet type.

[0012] In some embodiments, the first packet type is an Institute of Electrical and Electronics Engineers (IEEE) 802.11ad packet type, and the second packet type is an IEEE 802.11ay packet type. In some embodiments, adding the phase shift to the STF comprises applying a linear phase rotation other than

$$e^{j\pi\frac{n}{2}}$$

to the STF. In some embodiments, adding the phase shift to the STF comprises applying a block rotation to the STF.

[0013] In accordance with an embodiment, a method is provided. The method includes receiving a wireless packet comprising a short training field (STF), determining cross-correlations between the STF and a first preamble component sequence from a first set of preamble component sequences associated with a first packet type and between the STF and a second preamble component sequence from a second set of preamble component sequences associated with a second packet type, the first preamble component sequence and the second preamble component sequence having a zero correlation zone (ZCZ) property, and determining that the wireless packet is the second packet type when there is a greater correlation between the STF and the second preamble component sequence than between the STF and the first preamble component sequence.

[0014] In some embodiments, the first packet type is an Institute of Electrical and Electronics Engineers (IEEE) 802.11ad packet type, and the second packet type is an IEEE 802.11ay packet type. In some embodiments, the ZCZ property is a ZCZ of at least 64 symbol time slots in width. In some embodiments, the first set of preamble component sequences comprise preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$. In some embodiments, the second set of preamble component sequences comprise preamble component sequences $G_{a_{128,new,1}}$ and $G_{b_{128,new,1}}$. In some embodiments, the second set of preamble component sequences comprise mutual ZCZ sequences. In some embodiments, the mutual ZCZ sequences comprise preamble component sequences $G_{a_{128,new,2}}$ and $G_{b_{128,new,2}}$. In some embodiments, the second set of preamble component sequences comprise quadrature phase shift keying (QPSK) modulated preamble component sequences. In some embodiments, the QPSK modulated preamble component sequences comprise preamble component sequences A_{128} and B_{128} .

[0015] In accordance with an embodiment, a method is provided. The method includes selecting a first Golay sequence from a first pair of Golay complementary sequences associated with first packet type, each Golay sequence of the first pair of Golay complementary sequences being zero correlation zone (ZCZ) sequences with each

Golay sequence of a second pair of Golay complementary sequences associated with a second packet type, and transmitting a wireless packet carrying a short training field (STF) that includes one or more instances of the first Golay sequence.

[0016] In some embodiments, the first pair of Golay complementary sequences comprise preamble component sequences $G_{a_{128,new,1}}$ and $G_{b_{128,new,1}}$. In some embodiments, the first pair of Golay complementary sequences comprise mutual ZCZ sequences. In some embodiments, the mutual ZCZ sequences comprise preamble component sequences $G_{a_{128,new,2}}$ and $G_{b_{128,new,2}}$. In some embodiments, the first pair of Golay complementary sequences comprise quadrature phase shift keying (QPSK) modulated Golay complementary sequences. In some embodiments, the QPSK modulated Golay complementary sequences comprise preamble component sequences A_{128} and B_{128} . In some embodiments, transmitting the wireless packet comprises transmitting a real portion of the QPSK modulated Golay complementary sequences on a first antenna and transmitting an imaginary portion of the QPSK modulated Golay complementary sequences on a second antenna.

BRIEF DESCRIPTION OF THE DRAWINGS

[0017] For a more complete understanding of the present invention, and the advantages thereof, reference is now made to the following description taken in conjunction with the accompanying drawings, in which:

[0018] FIG. 1 is a diagram of an embodiment wireless communications network;

[0019] FIG. 2 is a diagram of a packet;

[0020] FIGS. 3A and 3B show preamble component sequences for PHY packets;

[0021] FIGS. 4A and 4B are diagrams of a non-control STF and a control STF;

[0022] FIGS. 5A-5D are diagrams of correlation properties of STFs with various preamble component sequences;

[0023] FIGS. 6A-8B illustrate a first auto-detection scheme using STFs;

[0024] FIGS. 9A-11B illustrate a second auto-detection scheme using STFs;

[0025] FIGS. 12A-16B illustrate a third auto-detection scheme using STFs;

[0026] FIGS. 17A-21B illustrate a fourth auto-detection scheme using STFs;

[0027] FIGS. 22A-25B illustrate a fifth auto-detection scheme using STFs;

[0028] FIG. 26 is a diagram of an embodiment processing system; and

[0029] FIG. 27 is a block diagram of a transceiver.

[0030] Corresponding numerals and symbols in the different figures generally refer to corresponding parts unless otherwise indicated. The figures are drawn to clearly illustrate the relevant aspects of the embodiments and are not necessarily drawn to scale.

DETAILED DESCRIPTION OF ILLUSTRATIVE EMBODIMENTS

[0031] The making and using of embodiments of this disclosure are discussed in detail below. It should be appreciated, however, that the concepts disclosed herein can be embodied in a wide variety of specific contexts, and that the specific embodiments discussed herein are merely illustrative.

tive and do not serve to limit the scope of the claims. Further, it should be understood that various changes, substitutions and alterations can be made herein without departing from the spirit and scope of this disclosure as defined by the appended claims.

[0032] IEEE 802.11ay is currently being developed as an NG60 extension to IEEE 802.11ad, with a goal of achieving extended throughput and range. Disclosed herein is a system and method for using short training fields to determine whether a received wireless physical (PHY) packet is an IEEE 802.11ay control packet, an IEEE 802.11ay non-control packet, an IEEE 802.11ad control packet, or an IEEE 802.11ad non-control packet.

[0033] In particular, STF fields in IEEE 802.11ay packets may be different from STF fields in IEEE 802.11ad packets such that a different numbers of cross-correlation peaks result when the respective STFs are cross-correlated with one or more preamble component sequences. As a result, an IEEE 802.11ay receiver may be able to distinguish IEEE 802.11ay packets from IEEE 802.11ad packets by performing cross-correlation. For example, the IEEE 802.11ay receiver may determine a number of cross-correlation peaks between an STF in a packet and a first preamble component sequence, and a number of cross-correlation peaks between the STF in the packet and a second preamble component sequence. The first preamble component sequence and the second preamble component sequence may be complementary sequences that have an out-of-phase aperiodic autocorrelation coefficients sum that is zero. In one embodiment, the first preamble component sequence and the second preamble component sequence form Golay pairs. The IEEE 802.11ay receiver may then determine whether the packet is an IEEE 802.11ay packet or an IEEE 802.11ad packet based on the numbers of correlation peaks between the STF in the packet and the first and second pairs of preamble component sequences. The IEEE 802.11ay receiver may determine that the packet is an IEEE 802.11ad packet when a number of cross-correlation peaks between the STF in the packet and the first preamble component sequence exceeds a first threshold (e.g., 16 peaks) or when a number of cross-correlation peaks between the STF in the packet and the second preamble component sequence exceeds a second threshold (e.g., 48 peaks). The IEEE 802.11ay receiver may determine that the packet is an IEEE 802.11ay packet when a number of cross-correlation peaks between the STF in the packet and the second preamble component sequence exceeds a first threshold (e.g., 16 peaks) or when a number of cross-correlation peaks between the STF in the packet and the first preamble sequence exceeds a second threshold (e.g., 48 peaks). Because embodiment 802.11ay packets can be distinguished from 802.11ad packets, 802.11ad receivers may discard 802.11ay packets, rather than continuing to process incompatible packets, which may reduce power consumption at the 802.11ad receivers. Such savings may be accomplished without reprogramming existing 802.11ad receivers, because such receivers already work in a way that allows them to discard embodiment 802.11ay packets.

[0034] Other techniques for distinguishing IEEE 802.11ay packets from IEEE 802.11ad packets are disclosed. In some embodiments, IEEE 802.11ay packets are transmitted using preamble component sequences that are phase-rotated variants of the preamble component sequences used to transmit IEEE 802.11ad packets. As a result, an IEEE 802.11ay receiver may be able to distinguish IEEE 802.11ay

packets from IEEE 802.11ad packets by determining a number of cross-correlation peaks between an STF in a packet and a pair of preamble component sequences, then removing a phase rotation from the STF and determining a number of cross-correlation peaks between the phase-rotated STF and the same pair of preamble component sequences. The IEEE 802.11ay receiver may then determine whether the packet is an IEEE 802.11ay packet or an IEEE 802.11ad packet according to whether the STF or the phase-rotated STF has a higher degree of cross-correlation with the preamble component sequences.

[0035] In some embodiments, preamble component sequences are selected for the STFs of control and non-control 802.11ay packets such that these preamble component sequences for 802.11ay are mutual zero correlation zone (ZCZ) sequences with respective preamble component sequences for 802.11ad. Preamble component sequences are sequences of symbols transmitted in the preamble (e.g., STF, CEF) of a frame that are known to a receiver and transmitter. The preamble component sequences in IEEE 802.11ad and 802.11ay are Golay complimentary sequences. In some embodiments, preamble component sequences are selected for the STFs of control and non-control packets in an 802.11ay packet such that these preamble component sequences are mutual zero correlation zone (ZCZ) sequences with a zero correlation zone in a range of about 65 time-domain symbol spacings. In some embodiments, the preamble component sequences for control and non-control 802.11ay packets are QPSK-modulated for use in 802.11ay packets.

[0036] Various embodiments may achieve advantages. By performing auto-detection in the STF, CEF, and/or header (s), a receiver may identify the format of a received packet at an early stage of reception. In particular, auto-detection can be performed very early in the reception pipeline in embodiment Physical Layer Convergence Protocol (PLCP) Protocol Data Units (PPDUs) that receive the STF before the CEF, header, or data fields. Additionally, proper selection of the STF and/or CEF of an 802.11ay packet may permit legacy 802.11ad devices to detect the length and modulation coding scheme (MCS) of 802.11ay packets, which may reduce ambiguity during auto-detection.

[0037] FIG. 1 is a diagram of a network **100** for communicating data. The network **100** comprises an access point (AP) **110** having a coverage area **101**, a plurality of mobile devices **120**, and a backhaul network **130**. As shown, the AP **110** establishes uplink (dashed line) and/or downlink (dotted line) connections with the mobile devices **120**, which serve to carry data from the mobile devices **120** to the AP **110** and vice-versa. Data carried over the uplink/downlink connections may include data communicated between the mobile devices **120**, as well as data communicated to/from a remote-end (not shown) by way of the backhaul network **130**. As used herein, the term “access point” refers to any component (or collection of components) configured to provide wireless access in a network, such as an evolved NodeB (eNB), a macro-cell, a femtocell, a Wi-Fi AP, or other wirelessly enabled devices. APs may provide wireless access in accordance with one or more wireless communication protocols, e.g., Long Term Evolution (LTE), LTE advanced (LTE-A), High Speed Packet Access (HSPA), Wi-Fi 802.11a/b/g/n/ac, etc. As used herein, the term “mobile device” refers to any component (or collection of components) capable of establishing a wireless connection

with an access point, such as a user equipment (UE), a mobile station (STA), and other wirelessly enabled devices. In some embodiments, the network **100** may comprise various other wireless devices, such as relays, low power nodes, etc. Embodiment techniques may be performed on a receiver in the network **100**, such as the AP **110**.

[0038] PPDUs are units of data transmitted over the physical (PHY) layer of a network, e.g., Layer 1 of the Open Systems Interconnection (OSI) model. PPDUs are structured data units that include information such as address information, protocol-control information, and/or user data. The packet structure of a PDU typically includes a short training field (STF), channel estimation field (CEF), header field, and data payload. Some PPDUs may also include a legacy header field (L-Header) and an NG60 header (N-Header) field. Auto-detection may be performed by detecting the preamble of an 802.11ad PDU, such as the STF and CEF, and/or the header(s). In particular, the STF and CEF in 802.11ad use structured bipolar-based complementary Golay sequences, which have several beneficial correlation properties that may be leverage by APs to enable auto-detection for 802.11ay networks.

[0039] FIG. 2 is a diagram of a packet **200**. The packet **200** may be a wireless PHY packet, and may be a control or non-control packet. Control PHY packets typically carry control information in their payloads, and non-control PHY packets typically carry data in their payloads. Non-control PHY packets may be transmitted using various waveforms, such as single carrier (SC) waveforms and orthogonal frequency-division multiplexed (OFDM) waveforms. A receiver may need to determine whether the packet **200** is a non-control or control PHY packet upon receiving the packet **200**. The receiver may also need to determine whether the packet **200** is an 802.11ad or an 802.11ay packet.

[0040] The packet **200** includes a STF **202**, a CEF **204**, a header **206**, a payload **208**, and training fields **210**. It should be appreciated that the packet **200** could include other fields. The STF **202** and the CEF **204** are commonly referred to in combination as a packet preamble **212**. In some embodiments, the STF **202** is used to determine whether the packet **200** is a control or non-control PHY packet, and whether the packet is an 802.11ad or an 802.11ay PHY packet.

[0041] The CEF **204** is used for channel estimation. If the packet **200** is a non-control packet, then the CEF **204** may also allow the receiver to determine what kind of waveform was used to communicate the packet **200**, e.g., an SC waveform or an OFDM waveform.

[0042] The header **206** may contain indicators or parameters that allow the receiver to decode the payload **208**. In some embodiments, the header **206** may be used to determine whether the packet is an 802.11ad PHY packet or an 802.11ay PHY packet.

[0043] The payload **208** contains information (e.g., data) carried by the packet **200**. The training fields **210** may include other fields such as automatic gain control (AGC) and training (TRN) R/T subfields appended to the packet **200**.

[0044] FIGS. 3A and 3B show preamble component sequences for PHY packets. The preamble component sequences are bipolar-based Golay complementary sequences having a length of 128 symbols. The preamble component sequences shown in FIG. 3A-B are referred to by IEEE 802.11ad as G_{128} and $G_{b_{128}}$, respectively. The

sequences $G_{a_{128}}$ and $G_{b_{128}}$ are Golay sequences that form a complementary pair, and the subscript indicates the sequence length of 128. The preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$ may be binary phase-shift keyed (BPSK), such that they are located at 0° and 180° on the unit circle, e.g., each symbol has a modulated value of either 1 or -1 . The preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$ may be transmitted in the STF of a PHY packet, such as the STF **202** of packet **200**.

[0045] FIGS. 4A and 4B are diagrams of a non-control STF **400** and a control STF **450**, respectively, which are included in an 802.11ad packet preamble. The non-control STF **400** and the control STF **450** each include repeated sequences **402**, **452** and a termination sequence **404**, **454**. The control STF **450** further includes a prefix sequence **456** after the termination sequence **454**.

[0046] The repeated sequences **402**, **452** are multiple repetitions of the preamble component sequences $G_{a_{128}}$ or $G_{b_{128}}$. The type and quantities of sequences in the repeated sequences **402**, **452** may be different between the non-control STF **400** and a control STF **450** so that a receiver may distinguish a non-control 802.11ad PHY packet from a control 802.11ad PHY packet. For example, the repeated sequence **402** may be 16 repetitions of the preamble component sequence $G_{a_{128}}$, and the repeated sequence **452** may be 48 repetitions of the preamble component sequence $G_{b_{128}}$.

[0047] The termination sequences **404**, **454** occur at the end of the repetition portion of the non-control STF **400** and the control STF **450**, respectively, and thus mark the end of the non-control STF **400** and the control STF **450**. As discussed above, an STF may include different values and have different lengths for a non-control or control PHY packet. As such, the termination sequences **404**, **454** are predetermined sequences that indicate the end of an STF. The termination sequences **404**, **454** are negative instances of the preamble component sequence used in the repeated sequence **402**, **452**, e.g., where each symbol in the termination sequences **404**, **454** is multiplied by -1 . For example, when the repeated sequence **452** is several repetitions of the preamble component sequence $G_{b_{128}}$, then the termination sequence **454** is a negated preamble component sequence $-G_{b_{128}}$. Accordingly, the preamble component sequences $-G_{a_{128}}$ and $-G_{b_{128}}$ may be chosen, respectively, for the termination sequences **404**, **454**.

[0048] The prefix sequence **456** occurs after the termination sequence **454** in the control STF **450**. The prefix sequence **456** is a preamble component sequence $-G_{a_{128}}$ and is used as the cyclic prefix for the CEF **204**. The termination sequence **404** of the non-control STF **400** also functions as a prefix sequence for the CEF **204**, because the termination sequence **404** is also the preamble component sequence $-G_{a_{128}}$.

[0049] FIGS. 5A and 5B are diagrams of correlation properties of the non-control STF **400** and the control STF **450** with the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$, respectively. A receiver may perform cross-correlation to determine whether a received sequence matches a known sequence. For example, a receiver may cross-correlate a received STF with the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$ to determine which preamble component sequence is carried within the STF.

[0050] As shown in FIG. 5A, when the non-control STF **400** is correlated with the preamble component sequence

$G_{a_{128}}$, 16 positive impulses and a negative impulse are generated. The 16 positive impulses correspond to correlation peaks with the 16 positive repetitions of the preamble component sequence $G_{a_{128}}$ in the repeated sequence **402**, and the negative impulse corresponds to correlation peaks with the negative instance of the preamble component sequence $G_{a_{128}}$ in the termination sequence **406**. The produced impulses are normalized to have a unit maximum magnitude, e.g., a maximum magnitude of 1 or -1 . As shown in FIG. 5B, when the non-control STF **400** is correlated with the preamble component sequence $G_{b_{128}}$, no correlation peaks are generated. Some noise may be generated, but the magnitude of the noise may not be large enough to register as a correlation peak. As shown in FIGS. 5A and 5B, while the magnitude of the noise is low, there is still an appreciable amount of noise present even when no correlation peak is generated. Accordingly, by cross-correlating an STF in a received packet with both of the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$, a receiver may be able to determine whether or not the packet is a non-control 802.11ad packet.

[0051] FIGS. 5C and 5D are diagrams of correlation properties of the control STF **450** with the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$, respectively. As shown in FIG. 5C, when the control STF **450** is correlated with the preamble component sequence $G_{a_{128}}$, one negative cross-correlation peak is generated. This is because, as shown above, the control STF **450** contains one negative instance of the preamble component sequence $G_{a_{128}}$. When the control STF **450** is correlated with the preamble component sequence $G_{b_{128}}$, 48 positive impulses and a negative impulse are generated. These impulses correspond to the 48 positive repetitions of the preamble component sequence $G_{b_{128}}$ in the repeated sequence **452** and the one negative instance of the preamble component sequence $G_{b_{128}}$ in the prefix sequence **456**. Accordingly, by cross-correlating an STF in a received packet with both of the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$, a receiver may be able to determine whether or not the packet is an 802.11ad control packet.

[0052] Embodiment receivers may also auto-detect receipt of an 802.11ay packet using the STF field. Accordingly, in some embodiments, STFs for 802.11ay packets may be selected such that they are different from STFs for 802.11ad packets. Embodiment receivers may then perform auto-detection of WLAN packets using the STF of received PHY packets. 802.11ay packets may be distinguished from 802.11ad packets using auto-detection schemes, according to autocorrelation and cross-correlation properties of the STFs with various preamble component sequences.

[0053] FIGS. 6A-8B illustrate a first auto-detection scheme using STFs in which the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$, which are used in 802.11ad STFs, may also be used in 802.11ay STFs. In order to distinguish 802.11ay packets from 802.11ad packets, the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$ may be interchanged between control and non-control STFs in 802.11ay such that $G_{a_{128}}$ is used for the repeated sequences in an 802.11ay control packet and $G_{b_{128}}$ is used for the repeated sequences in an 802.11ay non-control packet.

[0054] FIGS. 6A and 6B are diagrams of a non-control STF **600** and a control STF **650**, respectively, which are included in an 802.11ay packet preamble. The non-control

STF **600** and the control STF **650** each include repeated sequences **602**, **652**, a termination sequence **604**, **654**, and a prefix sequence **606**, **656**.

[0055] The repeated sequence **602** of the non-control STF **600** includes 16 repetitions of the preamble component sequence $G_{b_{128}}$, and the control STF **650** includes 48 repetitions of the preamble component sequence $G_{a_{128}}$. The termination sequence **604**, **654** in each STF includes a negative instance of the preamble component sequence used in the repeated sequences **602**, **652**, i.e., the sequence **604** has one repetition of $-G_{b_{128}}$, and the sequence **654** has one repetition of $-G_{a_{128}}$.

[0056] In some embodiments, the prefix sequence **606**, **656** is a predefined sequence, such as $-G_{a_{128}}$, used above in the non-control STF **400** and the control STF **450**. Adding a single repetition of $-G_{a_{128}}$ to the control STF **650** may assist in degrading channel estimation for an 802.11ad receiver that receives an 802.11ay PHY packet. An 802.11ad receiver may incorrectly identify a received control STF **650** as a non-control STF after detecting the repeated sequences **652** and the termination sequence **654**. Adding a second repetition of $-G_{a_{128}}$ after the termination sequence **654** prevents the 802.11ad receiver from successfully detecting the CEF **204**.

[0057] FIGS. 7A-7D are diagrams of cross-correlation properties for the non-control STF **600** and the control STF **650** with the preamble component sequences $G_{b_{128}}$ and $G_{a_{128}}$. FIG. 7A is a diagram of cross-correlation properties of the non-control STF **600** with the preamble component sequence $G_{a_{128}}$. FIG. 7B is a diagram of cross-correlation properties of the non-control STF **600** with the preamble component sequence $G_{b_{128}}$. FIG. 7C is a diagram of cross-correlation properties of the control STF **650** with the preamble component sequence $G_{a_{128}}$. FIG. 7D is a diagram of cross-correlation properties of the control STF **650** with the preamble component sequence $G_{b_{128}}$.

[0058] As shown in FIGS. 7A-7D and FIGS. 5A-5D, a relationship exists between the PHY packet type of a received STF, and a number of peaks generated when the received STF is cross-correlated with different preamble component sequences. This relationship can be expressed as a lookup table, as shown below in Table 1.

TABLE 1

Preamble Seq.	Qty. Peaks	Result	Reference
$G_{a_{128}}$	16	802.11ad; non-control	FIG. 5A
$G_{b_{128}}$	48	802.11ad; control	FIG. 5D
$G_{b_{128}}$	16	802.11ay; non-control	FIG. 7B
$G_{a_{128}}$	48	802.11ay; control	FIG. 7C

[0059] As shown above in Table 1, a received PHY packet type can be determined by cross-correlating the STF with the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$ and counting the number of positive peaks in the correlation result. Also included in Table 1 is a reference to the Figure number that shows a typical cross-correlation result for particular packet and frame types.

[0060] FIG. 8A is a diagram of an embodiment 802.11ay auto-detection method **800**. The 802.11ay auto-detection method **800** may be indicative of operations occurring in an 802.11ay receiver when determining the PHY packet type of a received packet using the STF.

[0061] The 802.11ay auto-detection method **800** begins by correlating a received STF with the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$ (step **802**). Next, the correlation peaks for each preamble component sequence are detected and counted (step **804**). If correlation of the received STF produces about 48 peaks for the preamble component sequence $G_{a_{128}}$ or about 16 peaks for the preamble component sequence $G_{b_{128}}$ (step **806**), the packet is identified as an 802.11ay packet (step **808**). In cases where correlation with one preamble component sequence produces peaks, correlation with the other sequence may produce other values that are not considered peaks, such as noise or zero values. For example, as shown in FIGS. 7C and 7D if 48 peaks are produced when correlated with the preamble component sequence $G_{a_{128}}$, then those 48 peaks will not be produced when correlated with the preamble component sequence $G_{b_{128}}$. Next, demodulation and decoding of the packet is continued until the end of the 802.11ay packet (step **810**).

[0062] If correlation of the STF produces 16 peaks for the preamble component sequence $G_{a_{128}}$ or 48 peaks for the preamble component sequence $G_{b_{128}}$ (step **812**), the packet is identified as an 802.11ad packet (step **814**). Finally, demodulation and decoding of the packet is continued until the end of the 802.11ad packet (step **816**).

[0063] FIG. 8B is a diagram of embodiment 802.11ad auto-detection method **850**. The 802.11ad auto-detection method **850** may be indicative of operations occurring in an 802.11ad receiver when determining the PHY packet type of a received packet using the STF. Because an 802.11ad receiver may not be capable of detecting 802.11ay packets, the 802.11ad auto-detection method **850** may thus avoid incorrectly identifying a received 802.11ay packet as a compatible packet for an 802.11ad receiver.

[0064] The 802.11ad auto-detection method **850** begins by receiving an 802.11ay STF and correlating it with the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$ (step **852**). Next, the correlation peaks for each preamble component sequence are detected and counted (step **854**). Next, the receiver determines whether correlation of the STF with the preamble component sequence $G_{a_{128}}$ produces 16 peaks (step **856**). The correlation with $G_{a_{128}}$ may not produce 16 peaks if, e.g., the STF is the non-control STF **600**. In this case, a detection failure will occur when attempting to count the correlation peaks for $G_{b_{128}}$ (step **858**). The detection failure may occur because an 802.11ad receiver expects correlation of a received STF with the preamble component sequence $G_{b_{128}}$ to produce 48 peaks (for a control STF) or no peaks (for a non-control STF). However, as shown in FIG. 7B, correlation of the non-control STF **600** with the preamble component sequence $G_{b_{128}}$ produces 16 peaks. Accordingly, an 802.11ad receiver produces a detection failure when attempting to detect the non-control STF **600**.

[0065] If, instead, correlation of the STF with the preamble component sequence $G_{a_{128}}$ produces 16 peaks, then the 802.11ad receiver will proceed to perform channel estimation and PHY header detection (step **860**). Such a state may occur when the STF is the control STF **650**. As shown in FIG. 7C, correlation of the control STF **650** with $G_{a_{128}}$ produces 48 peaks; however, some 802.11ad receivers may detect the first 16 peaks of the 48 correlated peaks and falsely identify the control STF **650** as the non-control STF **400**. In this case, detection failure will occur when performing PHY header detection (step **862**). When a detection failure for either the PHY header or STF occurs, then the

802.11ad auto-detection method **850** terminates. Accordingly, use of the non-control STF **600** and the control STF **650** in an 802.11ay PHY packet may allow an 802.11ad receiver to detect and discard 802.11ay PHY packets, notwithstanding the fact that an 802.11ad receiver may have no knowledge of the STF layout for 802.11ay packets.

$$(r_1, r_2, \dots, r_{2176}) r_n \in \{1, -1\} n=1, 2, \dots, 2176 (r_1 e^{j\theta_1}, r_2 e^{j\theta_2}, \dots, r_{2176} e^{j\theta_{2176}}) \theta_n \in (0, 2\pi) n=1, 2, \dots, 2176$$

FIGS. 9A-11B illustrate a second auto-detection scheme using STFs in which the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$, which are used in 802.11ad STFs, may be used in 802.11ay STFs. In order to distinguish 802.11ay packets from 802.11ad packets, a phase rotation may be applied to each bipolar symbol of the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$. The values in the repeated sequence **402** of the non-control STF **400** can be expressed generally as:

$$r_n \in \{1, -1\} n=1, 2, \dots, 2176 \theta_n \in (0, 2\pi) n=1, 2, \dots, 2176$$

where and. Thus, when applying a phase rotation to each bipolar symbol, the values in the repeated sequence for an 802.11ay non-control STF may be expressed generally as:

$$r_n \in \{1, -1\} n=1, 2, \dots, 2176 \theta_n \in (0, 2\pi) n=1, 2, \dots, 2176$$

where and

$$r_n \in \{1, -1\} n=1, 2, \dots, 2176 \theta_n \in (0, 2\pi) n=1, 2, \dots, 2176$$

$$r_n \in \{1, -1\} n=1, 2, \dots, 2176 \theta_n \in (0, 2\pi) n=1, 2, \dots, 2176$$

$$(r_1, r_2, \dots, r_{6400}) r_n \in \{1, -1\} n=1, 2, \dots, 6400 (r_1 e^{j\theta_1}, r_2 e^{j\theta_2}, \dots, r_{6400} e^{j\theta_{6400}}) \theta_n \in (0, 2\pi) n=1, 2, \dots, 6400$$

Likewise, the values in the repeated sequence **452** of the control STF **450** may be expressed generally as:

$$r_n \in \{1, -1\} n=1, 2, \dots, 6400 \theta_n \in (0, 2\pi) n=1, 2, \dots, 6400$$

where and. Thus, when applying a phase rotation to each bipolar symbol, the values in the repeated sequence for an 802.11ay control STF may be expressed generally as:

$$r_n \in \{1, -1\} n=1, 2, \dots, 6400 \theta_n \in (0, 2\pi) n=1, 2, \dots, 6400 \text{ where}$$

$$r_n \in \{1, -1\} n=1, 2, \dots, 6400 \theta_n \in (0, 2\pi) n=1, 2, \dots, 6400$$

$$r_n \in \{1, -1\} n=1, 2, \dots, 6400 \theta_n \in (0, 2\pi) n=1, 2, \dots, 6400 \text{ and.}$$

$$e^{j\frac{\pi}{2}} r_{STF}(nT_c) =$$

$$\begin{cases} G_{a_{128}} * (n \bmod 128) * e^{j\frac{\pi}{2}}, n = 0, 1, \dots, 16 * 128 - 1 \\ -G_{a_{128}} * (n \bmod 128) * e^{j\frac{\pi}{2}}, n = 16 * 128, \dots, 18 * 128 - 1 \end{cases}$$

$$r_{STF}(nT_c) =$$

$$\begin{cases} G_{b_{128}} * (n \bmod 128) * e^{j\frac{\pi}{2}}, n = 0, 1, \dots, 48 * 128 - 1 \\ -G_{b_{128}} * (n \bmod 128) * e^{j\frac{\pi}{2}}, n = 48 * 128, \dots, 49 * 128 - 1 e^{j\theta_n} \theta_n \neq \frac{\pi}{2} e^{j\theta_n} \varphi \neq \frac{\pi}{2} \varphi = -G_{a_{128}} * (n \bmod 128) * e^{j\frac{\pi}{2}}, n = 49 * 128, \dots, 50 * 128 - 1 \end{cases}$$

-continued

$$\frac{\pi}{4}, \frac{3\pi}{4}, \frac{\pi}{8}, \frac{3\pi}{8}, \frac{5\pi}{8},$$

[0066] As discussed above, each symbol in the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$ has a modulated value of either 1 or -1 . Thus each value is linearly phase-rotated on the unit circle by values other than π . For example, the values in the repeated sequence 402 of the non-control STF 400 may be expressed generally as:

$$e^{jn\frac{\pi}{2}} e^{j\theta_n} \theta_n \neq \frac{n\pi}{2} e^{jn\varphi} \varphi \neq \frac{\pi}{2} \varphi = \frac{\pi}{4}, \frac{3\pi}{4}, \frac{\pi}{8}, \frac{3\pi}{8}, \frac{5\pi}{8},$$

Likewise, the values in the repeated sequence 452 of the control STF 450 may be expressed generally as:

$$e^{jn\frac{\pi}{2}} e^{j\theta_n} \theta_n \neq \frac{n\pi}{2} e^{jn\varphi} \varphi \neq \frac{\pi}{2} \varphi = \frac{\pi}{4}, \frac{3\pi}{4}, \frac{\pi}{8}, \frac{3\pi}{8}, \frac{5\pi}{8},$$

In other words, a different linear phase rotation, may be selected and applied to the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$, e.g., π . Accordingly, in some embodiments, a linear phase rotation is used for 802.11ay, where, e.g., etc. A receiver may determine whether a received packet is an 802.11ad or an 802.11ay packet by cross-correlating the STF of the packet with the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$, and then removing a phase rotation from the STF of the packet and cross-correlating the de-rotated STF with the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$. The receiver may then determine the packet type according to which STF has a greater correlation with one or both of the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$, e.g., which STF produces an expected number of peaks.

$$e^{jn\frac{\pi}{2}} e^{j\theta_n} \theta_n \neq \frac{n\pi}{2} e^{jn\varphi} \varphi \neq \frac{\pi}{2} \varphi = \frac{\pi}{4}, \frac{3\pi}{4}, \frac{\pi}{8}, \frac{3\pi}{8}, \frac{5\pi}{8},$$

$$e^{jn\frac{\pi}{2}} e^{j\theta_n} \theta_n \neq \frac{n\pi}{2} e^{jn\varphi} \varphi \neq \frac{\pi}{2} \varphi = \frac{\pi}{4}, \frac{3\pi}{4}, \frac{\pi}{8}, \frac{3\pi}{8}, \frac{5\pi}{8},$$

[0067] FIG. 9A is a diagram of an embodiment 802.11ay auto-detection method 900. The 802.11ay auto-detection method 900 may be indicative of operations occurring on an 802.11ay receiver when determining the PHY packet type of a received packet using the STF.

[0068] The 802.11ay auto-detection method 900 begins by de-rotating the phase of a received STF by the phase shift used in IEEE 802.11ad, e.g.,

$$e^{-jn\frac{\pi}{2}}$$

(step 902). Such a de-rotation may be done to first determine whether the received STF is a BPSK-modulated symbol, e.g., an 802.11ad PHY STF. Next, the de-rotated STF is correlated with the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$ (step 904). Next, the correlation peaks for each preamble component sequence are detected and counted (step 906). If correlation of the received STF produces 16

peaks for the preamble component sequence $G_{a_{128}}$ or 48 peaks for the preamble component sequence $G_{b_{128}}$ (step 908), the packet is identified as an 802.11ad packet (step 910). Demodulation and decoding of the packet is continued until the end of the 802.11ad packet (step 912).

[0069] If correlation of the received STF does not produce 16 peaks for the preamble component sequence $G_{a_{128}}$ or 48 peaks for the preamble component sequence $G_{b_{128}}$, then the phase of the received STF is instead de-rotated by the phase shift used in IEEE 802.11ay, e.g., $e^{-jn\varphi}$ (step 914), where

$$\varphi \neq \frac{\pi}{2}.$$

Next, the de-rotated STF is correlated with the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$ (step 916). Next, the correlation peaks for each preamble component sequence are detected and counted (step 918). If correlation of the received STF produces 16 peaks for the preamble component sequence $G_{a_{128}}$ or 48 peaks for the preamble component sequence $G_{b_{128}}$ (step 920), the packet is identified as an 802.11ay packet (step 922). Finally, demodulation and decoding of the packet is continued until the end of the 802.11ay packet (step 924).

[0070] FIG. 9B is a diagram of an embodiment 802.11ad auto-detection method 950. The 802.11ad auto-detection method 950 may be indicative of operations occurring in an 802.11ad receiver.

[0071] The 802.11ad auto-detection method 950 begins by de-rotating the phase of a received 802.11ay STF by

$$e^{-jn\frac{\pi}{2}}$$

(step 952). The phase is de-rotated by

$$e^{-jn\frac{\pi}{2}}$$

because 802.11ad PHY packets contain BPSK-modulated symbols in their STFs, and so an 802.11ad receiver may only be capable of demodulating BPSK-modulated STFs. Next, the received STF is correlated with the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$ (step 954). Next, an attempt is made to detect and count the correlation peaks for each preamble component sequence (step 956). Finally, a failure occurs when attempting to count the correlation peaks. The failure occurs because the received 802.11ay STF may be modulated with a scheme other than BPSK, e.g., the symbols in the preamble component sequence may be linearly phase-rotated. An 802.11ad receiver may not detect these symbols. Accordingly, embodiment STFs may not interfere with an existing 802.11ad receiver.

[0072] In some embodiments, a block phase rotation $e^{j\theta_n}$ may be selected and applied to the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$ for an 802.11ay STF. That is, each of the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$ may be block-rotated to produce preamble component sequences $G_{a_{128,k}}$ and $G_{b_{128,k}}$, where k is a value from 1 to 17 for non-control STFs, and from 1 to 50 for control STFs.

[0073] FIGS. 10A and 10B are diagrams of a non-control STF 1000 and a control STF 1050, respectively, which are included in an 802.11ay packet preamble. As shown, each symbol in the preamble component sequences for the non-control STF 400 may be rotated by $e^{j\gamma_k}$, where $\gamma_k=k\pi$ for $k=1, \dots, 16$, and $\gamma_{17}=0$. Likewise, each symbol in the preamble component sequences for the control STF 450 may be rotated by $e^{j\phi_k}$, where $\phi_k=k\pi$ for $k=1, \dots, 48$, and $\phi_{49}=\phi_{50}=0$. Thus, each preamble component sequence in the repeated sequences 1002 may be represented by $Ga_{128,k}e^{j\gamma_k}$, where $\gamma_k \in \{0, 2\pi\}$, and each preamble component sequence in the repeated sequences 1052 may be represented by $Gb_{128,k}e^{j\phi_k}$, where $\phi_k \in \{0, 2\pi\}$.

[0074] FIG. 11A is a diagram of an embodiment 802.11ay auto-detection method 1100. The 802.11ay auto-detection method 1100 may be indicative of operations occurring in an 802.11ay receiver.

[0075] The 802.11ay auto-detection method 1100 beings by de-rotating the phase of a received STF by

$$e^{-j\pi\frac{n}{2}}$$

(step 1102). Such a de-rotation may be done to first determine whether the received STF is a BPSK-modulated symbol, e.g., an 802.11ad PHY STF. Next, the de-rotated STF is correlated with the preamble component sequences Ga_{128} and Gb_{128} (step 1104). Next, the correlation peaks for each preamble component sequence are detected and counted (step 1106). If correlation of the received STF produces 16 peaks for the preamble component sequence Ga_{128} or 48 peaks for the preamble component sequence Gb_{128} (step 1108), the packet is identified as an 802.11ad packet (step 1110). Demodulation and decoding of the packet is continued until the end of the 802.11ad packet (step 1112).

[0076] If correlation of the received STF does not produce 16 peaks for the preamble component sequence Ga_{128} or 48 peaks for the preamble component sequence Gb_{128} , then the phase of the received STF is instead block de-rotated by $e^{-j\gamma_k}$ for a non-control STF, or by $e^{-j\phi_k}$ for a control STF (step 1114). Next, the block de-rotated STF is correlated with the preamble component sequences Ga_{128} and Gb_{128} (step 1116). Next, the correlation peaks for each preamble component sequence are detected and counted (step 1118). If correlation of the received STF produces 16 peaks for the preamble component sequence Ga_{128} or 48 peaks for the preamble component sequence Gb_{128} (step 1120), the packet is identified as an 802.11ay packet (step 1122). Finally, demodulation and decoding of the packet is continued until the end of the 802.11ay packet (step 1124).

[0077] FIG. 11B is a diagram of an embodiment 802.11ad auto-detection method 1150. The 802.11ad auto-detection method 1150 may be indicative of operations occurring in an 802.11ad receiver.

[0078] The 802.11ad auto-detection method 1150 begins by de-rotating the phase of a received 802.11ay STF by

$$e^{-j\pi\frac{n}{2}}$$

(step 1152). The phase is de-rotated by

$$e^{-j\pi\frac{n}{2}}$$

because 802.11ad receivers may only be capable of decoding BPSK-modulated symbols in an STF. Next, the received STF is correlated with the preamble component sequences Ga_{128} and Gb_{128} (step 1154). Next, an attempt is made to detect and count the correlation peaks for each preamble component sequence (step 1156). Finally, a failure occurs when attempting to count the correlation peaks. The failure occurs because the received 802.11ay STF may be modulated with a scheme other than BPSK, e.g., the symbols in the preamble component sequence may be block phase-rotated. An 802.11ad receiver may not detect these symbols. Accordingly, embodiment STFs may not interfere with an existing 802.11ad receiver.

[0079] FIGS. 12A-16B illustrate a third auto-detection scheme using STFs in which new preamble component sequences are selected for the control and non-control STFs in an 802.11ay PHY packet. In some embodiments, new preamble component sequences $Ga_{128,new,1}$ and $Gb_{128,new,1}$ are selected such that they are zero correlation zone (ZCZ) sequences with Ga_{128} and Gb_{128} , respectively. Ideal ZCZ sequences have an impulse signal (e.g., values of 1 when in-phase and 0 when out-of-phase) when autocorrelated, and a value of 0 when cross-correlated in a certain range of shifts.

[0080] FIGS. 12A and 12B show embodiment preamble component sequences $Ga_{128,new,1}$ and $Gb_{128,new,1}$. The ZCZ properties of the preamble component sequences $Ga_{128,new,1}$ and $Gb_{128,new,1}$ may allow the sequences to be used as replacements for the preamble component sequences Ga_{128} and Gb_{128} in the repeated sequences 402 and 452 of the non-control STF 400 and the control STF 450, respectively.

[0081] FIGS. 13A-13H are diagrams of various correlation properties of the preamble component sequences $Ga_{128,new,1}$ and $Gb_{128,new,1}$ with the preamble component sequences Ga_{128} and Gb_{128} . FIGS. 13A and 13B are diagrams of autocorrelation properties of $Ga_{128,new,1}$ and $Gb_{128,new,1}$, respectively. As shown, the autocorrelation results are impulse responses for shifts within $[-32, 32]$ symbol time slots, and have no correlation for shifts within $[-32, 32]$ symbol time slots. Further, the side lobes for each autocorrelation signal are relatively low. Both the impulse response and the side lobes that result from autocorrelation of $Ga_{128,new,1}$ and $Gb_{128,new,1}$ are comparable to the impulse response and the side lobes that result from autocorrelation of Ga_{128} and Gb_{128} , previously shown in the zoomed portions of FIGS. 5A and 5C. It should be appreciated that, although FIGS. 13A-13H illustrate ideal impulse responses, autocorrelation results may not produce true impulses, but rather may produce responses that are of sufficient magnitude for a receiver to recognize the values as correlation peaks.

[0082] FIGS. 13C and 13D are diagrams of cross-correlation properties of $Ga_{128,new,1}$ and $Gb_{128,new,1}$, respectively. The cross-correlation properties of $Ga_{128,new,1}$ and $Gb_{128,new,1}$ are comparable to the cross-correlations between Ga_{128} and Gb_{128} .

[0083] FIGS. 13E and 13F are diagrams of cross-correlation properties between $Ga_{128,new,1}$ and Ga_{128} , and between $Gb_{128,new,1}$ and Gb_{128} , respectively. As shown, the cross-correlations are equal to zero for shifts within $[-32, 32]$. Unlike the cross-correlation between Ga_{128} and Gb_{128} (illus-

trated above in FIG. 5B), there is no noise in the zero correlation zone range for $G_{a_{128,new,1}}$ and $G_{b_{128,new,1}}$. Accordingly, an 802.11ad receiver will not detect a false positive when attempting to correlate $G_{a_{128}}$ or $G_{b_{128}}$ with $G_{a_{128,new,1}}$ or $G_{b_{128,new,1}}$, respectively.

[0084] FIGS. 13G and 13H are diagrams of cross-correlation properties between $G_{a_{128,new,1}}$ and $G_{b_{128}}$, and between $G_{b_{128,new,1}}$ and $G_{a_{128}}$, respectively. The cross-correlation properties between $G_{a_{128,new,1}}$ and $G_{b_{128}}$, and between $G_{b_{128,new,1}}$ and $G_{a_{128}}$ are comparable to the cross-correlations between $G_{a_{128}}$ and $G_{b_{128}}$.

[0085] FIGS. 14A and 14B are diagrams of a non-control STF 1400 and a control STF 1450, respectively, which are included in an 802.11ay packet preamble. The non-control STF 1400 and the control STF 1450 each include repeated sequences 1402, 1452, a termination sequence 1404, 1454 which indicates the end of the non-control STF 1400 and the control STF 1450, respectively, and a prefix sequence 1406, 1456.

[0086] The repeated sequences 1402, 1452 include multiple repetitions of preamble component sequences. The repeated sequence 1402 includes 16 repetitions of the preamble component sequence $G_{a_{128,new,1}}$, and the repeated sequence 1452 includes 48 repetitions of the preamble component sequence $G_{b_{128,new,1}}$. It should be appreciated that the repeated sequences 1402, 1452 could contain any preamble component sequence and number of repetitions.

[0087] The termination sequence 1404, 1454 occurs after the repeated sequences 1402, 1452. The termination sequence 1404, 1454 includes one negative instance of the preamble component sequences included in the repeated sequences 1402, 1452. For example, in embodiments where the repeated sequence 1402 includes several repetitions of the preamble component sequence $G_{a_{128,new,1}}$, the termination sequence 1404 includes one repetition of $-G_{a_{128,new,1}}$. Likewise, in embodiments where the repeated sequence 1452 includes several repetitions of the preamble component sequence $G_{b_{128,new,1}}$, the sequence 1454 includes one repetition of $-G_{b_{128,new,1}}$.

[0088] The prefix sequence 1406, 1456 is a cyclic prefix of CEF. It should be noted that the prefix sequence 1406, 1456, i.e., the final preamble component sequence ($-G_{a_{128}}$) before the CEF, may be the same preamble component sequence at the end of the non-control STF 400 and the control STF 450.

[0089] FIGS. 15A-15D are diagrams of cross-correlation properties for the non-control STF 1400 with various preamble component sequences. As shown, the non-control STF 1400 produces 16 positive peaks and a negative peak when correlated with the preamble component sequence $G_{a_{128,new,1}}$, and produces one negative peak when correlated with the preamble component sequence $G_{a_{128}}$, but otherwise does not produce a significant signal when correlated with other preamble component sequences. The values produced by cross-correlations of the non-control STF 1400 with sequences other than $G_{a_{128,new,1}}$ and $G_{a_{128}}$ may include some noise, but the magnitude of the noise may not be large enough to register as a correlation peak.

[0090] FIGS. 15E-15H are diagrams of cross-correlation properties for the control STF 1450 with various preamble component sequences. As shown, the control STF 1450 produces 48 positive peaks and a negative peak when correlated with the preamble component sequence $G_{b_{128,new,1}}$, and produces one negative peak when correlated with preamble component sequence $G_{b_{128}}$, but otherwise does not

produce a significant signal. The values produced by cross-correlations of the control STF 1450 with sequences other than $G_{b_{128,new,1}}$ may include some noise, but the magnitude of the noise may not be large enough to register as a correlation peak. Accordingly, proper selection of the symbols for the preamble component sequences used in the non-control STF 1400 and the control STF 1450 may allow the STFs to be auto-detected by a receiver that correlates the STFs with the preamble component sequences $G_{a_{128,new,1}}$ and $G_{b_{128,new,1}}$ (e.g., an 802.11ay receiver), but may not interfere with a receiver that correlates the STFs with the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$ (e.g., an 802.11ad receiver).

[0091] FIG. 16A is a diagram of an embodiment 802.11ay auto-detection method 1600. The 802.11ay auto-detection method 1600 may be indicative of operations occurring in an 802.11ay receiver when determining the PHY packet type of a received packet using the STF.

[0092] The 802.11ay auto-detection method 1600 begins by receiving a STF and correlating it with the preamble component sequences $G_{a_{128,new,1}}$, $G_{b_{128,new,1}}$, $G_{a_{128}}$, and $G_{b_{128}}$ (step 1602). Next, the correlation peaks for each preamble component sequence are detected and counted (step 1604). If correlation of the received STF produces 16 peaks for the preamble component sequence $G_{a_{128,new,1}}$ or 48 peaks for the preamble component sequence $G_{b_{128,new,1}}$ (step 1606), the packet is identified as an 802.11ay packet (step 1608). In cases where correlation with one preamble component sequence produces peaks, correlation with the other sequence may produce lower values that have no significance. For example, if 16 peaks are produced when correlated with the preamble component sequence $G_{a_{128,new,1}}$, then no values are produced when correlated with the preamble component sequence $G_{b_{128,new,1}}$. Next, demodulation and decoding of the packet is continued until the end of the 802.11ay packet (step 1610).

[0093] If instead, correlation of the STF produces 16 peaks for the preamble component sequence $G_{a_{128}}$ or 48 peaks for the preamble component sequence $G_{b_{128}}$ (step 1612), the packet is identified as an 802.11ad packet (step 1614). Finally, demodulation and decoding of the packet is continued until the end of the 802.11ad packet (step 1616).

[0094] FIG. 16B is a diagram of an embodiment 802.11ad auto-detection method 1650. The 802.11ad auto-detection method 1650 may be indicative of operations occurring in an 802.11ad receiver when determining the PHY packet type of a received packet using the STF. Because an 802.11ad receiver may not be capable of responding to 802.11ay traffic, the 802.11ad auto-detection method 1650 may thus avoid incorrectly identifying a received 802.11ay packet as a compatible packet for an 802.11ad receiver.

[0095] The 802.11ad auto-detection method 1650 begins by receiving an 802.11ay STF and correlating it with the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$ (step 1652). Next, the correlation peaks for each preamble component sequence are detected and counted (step 1654). Next, an attempt to detect positive peaks in the correlated preamble component sequences is attempted, but fails (step 1656). Detection fails because the received STFs contain the preamble component sequence $G_{a_{128,new,1}}$ and $G_{b_{128,new,1}}$, which are ZCZ sequences with $G_{a_{128}}$ and $G_{b_{128}}$, as shown in FIGS. 13E and 13F, and thus correlation does not produce significant values. Finally, the failed STF detection causes a synchronization failure (step 1658). Thus, the 802.11ad

auto-detection method **1650** allows an 802.11ad receiver to fail auto-detection when it receives an incompatible 802.11ay PHY packet, causing the receiver to ignore the remainder of the PHY packet.

[**0096**] FIGS. **17A-21B** illustrate a fourth auto-detection scheme using STF's in which new preamble component sequences are selected for the control and non-control STF's in an 802.11ay PHY packet. In some embodiments, new preamble component sequences $G_{a_{128,new,2}}$ and $G_{b_{128,new,2}}$ are selected such that they are mutual zero correlation zone (ZCZ) sequences, e.g., the sequences produce no correlation peaks when cross-correlated in a certain range of shifts.

[**0097**] FIGS. **17A** and **17B** show embodiment preamble component sequences $G_{a_{128,new,2}}$ and $G_{b_{128,new,2}}$. The mutual ZCZ properties of the preamble component sequences $G_{a_{128,new,2}}$ and $G_{b_{128,new,2}}$ allows the sequences to be used as replacements for the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$ in the repeated sequences **402**, **452** of the non-control STF **400** and the control STF **450**. Because the preamble component sequences $G_{a_{128,new,2}}$ and $G_{b_{128,new,2}}$ produce no correlation peaks when cross-correlated, there may be a reduced possibility of incorrectly identifying a PHY packet as a control or non-control packet based on the STF.

[**0098**] FIG. **18A-18C** are diagrams of various correlation properties of the preamble component sequences $G_{a_{128,new,2}}$ and $G_{b_{128,new,2}}$. FIGS. **18A** and **18B** show autocorrelation properties of $G_{a_{128,new,2}}$ and $G_{b_{128,new,2}}$, respectively. As shown, the autocorrelations have an impulse response property. Further, values at the side lobes of each autocorrelation are relatively low, and may be even lower than values at the side lobes of autocorrelations for the preamble component sequences $G_{a_{128,new,2}}$ and $G_{b_{128,new,2}}$ (shown in FIGS. **13A** and **13B**). Both the impulse response and side lobes for $G_{a_{128,new,2}}$ and $G_{b_{128,new,2}}$ are comparable to the impulse response and side lobes for $G_{a_{128}}$ and $G_{b_{128}}$, previously shown in the zoomed portions of FIGS. **5A** and **5C**.

[**0099**] FIG. **18C** is a diagram of a cross-correlation property of $G_{a_{128,new,2}}$ and $G_{b_{128,new,2}}$. $G_{a_{128,new,2}}$ and $G_{b_{128,new,2}}$ have a higher degree of correlation than the cross-correlation between $G_{a_{128}}$ and $G_{b_{128}}$. As shown in FIG. **18C**, the cross-correlation of the preamble component sequences $G_{a_{128,new,2}}$ and $G_{b_{128,new,2}}$ produces relatively low values.

[**0100**] FIGS. **19A** and **19B** are diagrams of a non-control STF **1900** and a control STF **1950**, respectively, which are included in an 802.11ay packet preamble. The non-control STF **1900** and the control STF **1950** each include repeated sequences **1902**, **1952**, a termination sequence **1904**, **1954**, and a prefix sequence **1906**, **1956**.

[**0101**] The repeated sequences **1902**, **1952** include multiple repetitions of the preamble component sequences. The repeated sequence **1902** includes 16 repetitions of the preamble component sequence $G_{a_{128,new,2}}$, and the repeated sequence **1952** includes 48 repetitions of the preamble component sequence $G_{b_{128,new,2}}$. It should be appreciated that the repeated sequences **1902**, **1952** could contain any preamble component sequence and number of repetitions, as discussed further below.

[**0102**] The termination sequence **1904**, **1954** occurs after the repeated sequences **1902**, **1952**. The sequence **1904**, **1954** includes one negative instance of the preamble component sequences included in the repeated sequences **1902**, **1952**. For example, in embodiments where the repeated

sequence **1902** includes several repetitions of the preamble component sequence $G_{a_{128,new,2}}$, the sequence **1904** includes one repetition of $-G_{a_{128,new,2}}$. Likewise, in embodiments where the repeated sequence **1952** includes several repetitions of the preamble component sequence $G_{b_{128,new,2}}$, the sequence **1954** includes one repetition of $-G_{b_{128,new,2}}$.

[**0103**] The prefix sequence **1906**, **1956** indicates the end of the non-control STF **1900** and the control STF **1950**, respectively. It should be noted that the prefix sequence **1906**, **1956**, i.e., the final preamble component sequence ($-G_{am}$) before the CEF, may be the same as the final sequence in the non-control STF **400** and the control STF **450**. Accordingly, an 802.11ad receiver may still be capable of identifying the end of an 802.11ay STF, even though an 802.11ad receiver may not be completely compatible with 802.11ay PHY packets. The prefix sequence **1906**, **1956** is a cyclic prefix for the CEF **204**.

[**0104**] FIGS. **20A-20D** show cross-correlation properties for the non-control STF **1900** and the control STF **1950** with the preamble component sequences $G_{a_{128,new,2}}$ and $G_{b_{128,new,2}}$. As shown, the non-control STF **1900** produces 16 positive peaks and a negative peak when correlated with the preamble component sequence $G_{a_{128,new,2}}$, but otherwise does not produce a significant signal when correlated with other preamble component sequences. Likewise, the control STF **1950** produces 48 positive peaks and a negative peak when correlated with the preamble component sequence $G_{b_{128,new,2}}$, but otherwise does not produce a significant signal when correlated with other preamble component sequences. The values produced by cross-correlations of the non-control STF **1900** and the control STF **1950** with sequences other than $G_{a_{128,new,2}}$ and $G_{b_{128,new,2}}$ may include some noise, but the magnitude of the noise may not be large enough to register as a correlation peak. Accordingly, proper selection of the symbols for the preamble component sequences used in the non-control STF **1900** and the control STF **1950** may allow the STF's to be auto-detected by a receiver that correlates the STF's with the preamble component sequences $G_{a_{128,new,2}}$ and $G_{b_{128,new,2}}$ (e.g., an 802.11ay receiver), but may not interfere with a receiver that correlates the STF's with the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$ (e.g., an 802.11ad receiver). Cross-correlation properties of the non-control STF **1900** and the control STF **1950** with other preamble component sequences (e.g., $G_{a_{128}}$, $G_{b_{128}}$) may be similar to properties discussed above with respect to FIGS. **15A-15H**. Because $G_{a_{128,new,2}}$ and $G_{b_{128,new,2}}$ are mutual ZCZ sequences, they produce less noise when cross-correlated than $G_{a_{128,new,1}}$ and $G_{b_{128,new,1}}$. For example, FIG. **20A** contains the same quantity of peaks as FIG. **15A**, but has less noise.

[**0105**] FIG. **21A** is a diagram of an embodiment 802.11ay auto-detection method **2100**. The 802.11ay auto-detection method **2100** may be indicative of operations occurring in an 802.11ay receiver when determining the PHY packet type of a received packet using the STF.

[**0106**] The 802.11ay auto-detection method **1600** begins by receiving a STF and correlating it with the preamble component sequences $G_{a_{128,new,2}}$, $G_{b_{128,new,2}}$, $G_{a_{128}}$, and $G_{b_{128}}$ (step **2102**). Next, the correlation peaks for each preamble component sequence are detected and counted (step **2104**). If correlation of the received STF produces 16 peaks for the preamble component sequence $G_{a_{128,new,2}}$ or 48 peaks for the preamble component sequence $G_{b_{128,new,2}}$ (step **2106**), the packet is identified as an 802.11ay packet

(step **2108**). In cases where correlation with one preamble component sequence produces peaks, correlation with the other sequence may produce lower values that have no significance. For example, if 16 peaks are produced when correlated with the preamble component sequence $G_{a_{128, new, 2}}$, then no values are produced when correlated with the preamble component sequence $G_{b_{128, new, 2}}$. Next, demodulation and decoding of the packet is continued until the end of the 802.11ay packet (step **2110**).

[0107] If instead, correlation of the STF produces 16 peaks for the preamble component sequence $G_{a_{128}}$ or 48 peaks for the preamble component sequence $G_{b_{128}}$ (step **2112**), the packet is identified as an 802.11ad packet (step **2114**). Finally, demodulation and decoding of the packet is continued until the end of the 802.11ad packet (step **2116**).

[0108] FIG. 21B is a diagram of an embodiment 802.11ad auto-detection method **2150**. The 802.11ad auto-detection method **2150** may be indicative of operations occurring in an 802.11ad receiver when determining the PHY packet type of a received packet using the STF. Because an 802.11ad receiver may not capable of detecting 802.11ay packet, the 802.11ad auto-detection method **2150** may thus avoid incorrectly identifying a received 802.11ay packet as a compatible packet for an 802.11ad receiver.

[0109] The 802.11ad auto-detection method **2150** begins by receiving an 802.11ay STF and correlating it with the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$ (step **2152**). Next, the correlation peaks for each preamble component sequence are detected and counted (step **2154**). Next, an attempt to detect positive peaks in the correlated preamble component sequences is attempted, but fails (step **2156**). Detection fails because the received STFs contain the preamble component sequence $G_{a_{128, new, 2}}$ and $G_{b_{128, new, 2}}$, which are mutual ZCZ sequences and do not correlate with $G_{a_{128}}$ and $G_{b_{128}}$, as shown in FIGS. 20C-20D and 20G-20H. Finally, the failed detection causes a synchronization failure (step **2158**). Thus, the 802.11ad auto-detection method **2150** allows an 802.11ad receiver to gracefully fail auto-detection and continue operating when it receives an incompatible 802.11ay PHY packet.

[0110] FIGS. 22A-25B illustrate a fifth auto-detection scheme using STFs in which new preamble component sequences are selected for the control and non-control STFs in an 802.11ay PHY packet. In some embodiments, new preamble component sequences A_{128} and B_{128} are selected such that the sequences are quadrature phase shift keying (QPSK) modulated variants of the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$.

$$\begin{aligned} A_{128} &= \alpha \cdot G_{a_{128}} + \beta \cdot G_{a_{128, new, 1}} \cdot e^{j\pi/2} B_{128} = \alpha \cdot G_{b_{128}} + \\ & \beta \cdot G_{b_{128, new, 1}} \cdot e^{j\pi/2} 0 < \alpha, \beta < 1, \alpha^2 + \\ & \beta^2 = 1, A_{128} = \alpha \cdot G_{a_{128}} + \beta \cdot G_{b_{128, new, 1}} \cdot e^{j\pi} \\ & 2B_{128} = \alpha \cdot G_{b_{128}} + \beta \cdot G_{a_{128, new, 1}} \cdot e^{j\pi/2} 0 < \alpha, \beta < 1, \alpha^2 + \\ & \beta^2 = 1 \end{aligned}$$

FIGS. 22A-22B show embodiment preamble component sequences A_{128} and B_{128} , respectively. The preamble component sequences A_{128} and B_{128} are chosen such that

$$0 < \alpha, \beta < 1, \alpha^2 + \beta^2 = 1 \text{ and } 0 < \alpha, \beta < 1, \alpha^2 + \beta^2 = 1 \text{ and}$$

$$0 < \alpha, \beta < 1, \alpha^2 + \beta^2 = 1 \text{ and } 0 < \alpha, \beta < 1, \alpha^2 + \beta^2 < 1$$

where. Thus, the preamble component sequences A_{128} and B_{128} are QPSK-modulated. This allows the preamble component sequences A_{128} and B_{128} to be backwards-compatible with other wireless techniques. In some embodiments, the

preamble component sequences $G_{a_{128, new, 1}}$ and $G_{b_{128, new, 1}}$ may be interchanged such that the imaginary portions of A_{128} and B_{128} are replaced with one another to produce preamble component sequences A_{128}' and B_{128}' such that

$$0 < \alpha, \beta < 1, \alpha^2 + \beta^2 = 1 \text{ and } 0 < \alpha, \beta < 1, \alpha^2 + \beta^2 = 1 \text{ and}$$

$$0 < \alpha, \beta < 1, \alpha^2 + \beta^2 = 1 \text{ and } 0 < \alpha, \beta < 1, \alpha^2 + \beta^2 < 1$$

where. It should be appreciated that in some embodiments A_{128} and B_{128} may be interchanged. Because the preamble component sequences A_{128} and B_{128} are QPSK-modulated variants of the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$, correlation of the preamble component sequences A_{128} and B_{128} with $G_{a_{128}}$ and $G_{b_{128}}$, respectively, may produce values that are equivalent to impulse responses in a certain range of shifts.

$$0 < \alpha, \beta < 1, \alpha^2 + \beta^2 = 1 \text{ and } 0 < \alpha, \beta < 1, \alpha^2 + \beta^2 = 1$$

$$0 < \alpha, \beta < 1, \alpha^2 + \beta^2 = 1 \text{ and } 0 < \alpha, \beta < 1, \alpha^2 + \beta^2 = 1$$

$$0 < \alpha, \beta < 1, \alpha^2 + \beta^2 = 1 \text{ and } 0 < \alpha, \beta < 1, \alpha^2 + \beta^2 = 1$$

$$0 < \alpha, \beta < 1, \alpha^2 + \beta^2 = 1 \text{ and } 0 < \alpha, \beta < 1, \alpha^2 + \beta^2 = 1$$

[0111] FIGS. 23A and 23B are diagrams of a non-control STF **2300** and a control STF **2350**, respectively, which are included in an 802.11ay packet preamble. The non-control STF **2300** and the control STF **2350** each include repeated sequences **2302**, **2352** and a termination sequence **2306**, **2356**. The control STF **2350** also includes a sequence **2354** interposed between the repeated sequence **2352** and the termination sequence **2356**. In some embodiments, the non-control STF **2300** may also include a sequence after the repeated sequence **2302**.

[0112] The repeated sequences **2302**, **2352** include multiple repetitions of the preamble component sequences. The repeated sequence **2302** includes 16 repetitions of the preamble component sequence A_{128} , and the repeated sequence **2352** includes 48 repetitions of the preamble component sequence B_{128} . It should be appreciated that the repeated sequences **2302**, **2352** could contain any preamble component sequence and number of repetitions.

[0113] The sequence **2354** occurs after the repeated sequence **2352**. The sequence **2354** includes one negative instance of the preamble component sequences included in the repeated sequence **2352**. For example, in embodiments where the repeated sequence **2352** includes several repetitions of the preamble component sequence B_{128} , the sequence **2354** includes one repetition of $-B_{128}$.

[0114] The termination sequence **2306**, **2356** is used as a cyclic prefix for the CEF **204** in the non-control STF **2300** and the control STF **2350**, respectively. It should be noted that the termination sequence **2306**, **2356**, i.e., the final preamble component sequence ($-G_{a_{128}}$) before the CEF, may be the same as the final preamble component sequence ($-G_{a_{128}}$) before the CEF the non-control STF **400** and the control STF **450**.

[0115] Because the preamble component sequences A_{128} and B_{128} are QPSK-modulated variants of the preamble component sequences $G_{a_{128}}$ and $G_{b_{128}}$, correlation of the non-control STF **2300** with $G_{a_{128}}$ by a receiver may produce 16 positive peaks and one negative peak, and correlation of the control STF **2350** with $G_{b_{128}}$ by a receiver may produce 48 positive peaks and one negative peak.

[0116] FIG. 24A is a diagram of an embodiment 802.11ay auto-detection method **2400**. The 802.11ay auto-detection

method **2400** may be indicative of operations occurring in an 802.11ay receiver when determining the PHY packet type of a received packet using the STF.

[**0117**] The 802.11ay auto-detection method **2400** begins by receiving a STF and correlating it with the preamble component sequences A_{128} , B_{128} , Ga_{128} , and Gb_{128} (step **2402**). Next, the correlation peaks for each preamble component sequence are detected and counted (step **2404**). If correlation of the received STF produces 16 peaks for the preamble component sequence Ga_{128} or 48 peaks for the preamble component sequence Gb_{128} (step **2406**), the packet is identified as one of an 802.11ay packet or an 802.11ad packet. If correlation peaks are not detected for either of the preamble component sequences Ga_{128} or Gb_{128} , then the 802.11ay auto-detection method **2400** concludes. Such a failure may occur, for example, when an 802.11ay receiver receives an incompatible PHY packet.

[**0118**] If correlation peaks are detected for either of the preamble component sequences Ga_{128} or Gb_{128} , the 802.11ay auto-detection method **2400** continues by determining whether correlation of the received STF produces 16 peaks for the preamble component sequence A_{128} or 48 peaks for the preamble component sequence B_{128} (step **2408**). If so, the packet is identified as an 802.11ay packet (step **2410**). Next, demodulation and decoding of the packet is continued until the end of the 802.11ay packet (step **2412**).

[**0119**] If correlation of the received STF does not produce 16 peaks for the preamble component sequence A_{128} or 48 peaks for the preamble component sequence B_{128} , then the packet is identified as an 802.11ad packet (step **2414**). Finally, demodulation and decoding of the packet is continued until the end of the 802.11ad packet (step **2416**). Accordingly, an 802.11ay receiver may be capable of detecting reception of a compatible PHY packet by correlating its STF with Ga_{128} and Gb_{128} , and then further correlating the received STF with A_{128} and B_{128} to auto-detect the received PHY packet type.

[**0120**] FIG. **24B** is a diagram of an embodiment 802.11ad auto-detection method **2450**. The 802.11ad auto-detection method **2450** may be indicative of operations occurring in an 802.11ad receiver when determining the PHY packet type of a received packet using the STF. Because an 802.11ad receiver may not be capable of detecting an 802.11ay packet, the 802.11ad auto-detection method **2450** may thus avoid incorrectly identifying a received 802.11ay packet as a compatible packet for an 802.11ad receiver.

[**0121**] The 802.11ad auto-detection method **2450** begins by receiving an 802.11ay STF and correlating it with the preamble component sequences Ga_{128} and Gb_{128} (step **2452**). Next, the correlation peaks for each preamble component sequence are detected and counted (step **2454**). If correlation of the received STF does not produce 16 peaks for the preamble component sequence Ga_{128} or 48 peaks for the preamble component sequence Gb_{128} (step **2456**), then auto-detection fails and the 802.11ad auto-detection method **2450** concludes.

[**0122**] However, if correlation of the received STF produces 16 peaks for the preamble component sequence Ga_{128} or 48 peaks for the preamble component sequence Gb_{128} (step **2456**), then channel estimation and L-Header detection of the received 802.11ay PHY packet is performed (step **2458**). Next, the PHY packet length and the MCS of the L-Header is detected (Step **2460**). Next, a detection failure occurs when detecting the N-Header because the 802.11ad

receiver is incompatible with an 802.11ay PHY packet (step **2462**). Finally, this failure causes the receiver to enter power save mode until the end of the current PHY packet is detected. Accordingly, an 802.11ad receiver may be capable of detecting reception of either an 802.11ay or an 802.11ad PHY packet during STF correlation, and then may gracefully fail auto-detection of an incompatible 802.11ay PHY packet after STF detection and during channel estimation and header detection. By failing auto-detection at a later stage, an 802.11ad receiver may be able to enter power save mode after receiving an incompatible PHY packet, increasing power efficiency of the receiver.

[**0123**] FIGS. **25A** and **25B** are diagrams of antenna transmission configurations **2500**, **2550** for a non-control STF and a control STF, respectively. The antenna transmission configuration **2500** may be indicative of signals transmitted to produce the non-control STF **2300**. Likewise, the antenna transmission configuration **2550** may be indicative of signals transmitted to produce the control STF **2350**. Because the preamble component sequences A_{128} and B_{128} are QPSK-modulated variants of the preamble component sequences Ga_{128} and Gb_{128} , an 802.11ay transmitter may produce the non-control STF **2300** and the control STF **2350** by transmitting orthogonal variants of the non-control STF **400** and the control STF **450**, respectively. The orthogonal sequences may be combined by transmitting each sequence on a different antenna. For example, a real portion of the preamble component sequences A_{128} and B_{128} may be transmitted on a first antenna, and an imaginary portion of the preamble component sequences A_{128} and B_{128} may be transmitted on a second antenna.

[**0124**] Because one of each of the transmitted sequences includes the preamble component sequences Ga_{128} and Gb_{128} , an 802.11ad receiver may thus receive the incompatible PHY packets and perform channel estimation and header detection of the PHY packet before failing auto-detection and entering power save mode. Likewise, an 802.11ay receiver may receive the PHY packet and perform auto-detection by correlating with the preamble component sequences A_{128} , B_{128} , Ga_{128} , and Gb_{128} . Auto-detection of the received PHY packets by a receiver may be performed using a method similar to the 802.11ay auto-detection method **2400** and the 802.11ad auto-detection method **2450**, discussed above.

[**0125**] FIG. **26** is a block diagram of an embodiment processing system **2600** for performing methods described herein, which may be installed in a host device. As shown, the processing system **2600** includes a processor **2602**, a memory **2604**, and interfaces **2606-2610**, which may (or may not) be arranged as shown in FIG. **26**. The processor **2602** may be any component or collection of components adapted to perform computations and/or other processing related tasks, and the memory **2604** may be any component or collection of components adapted to store programming and/or instructions for execution by the processor **2602**. In an embodiment, the memory **2604** includes a non-transitory computer readable medium. The interfaces **2606**, **2608**, **2610** may be any component or collection of components that allow the processing system **2600** to communicate with other devices/components and/or a user. For example, one or more of the interfaces **2606**, **2608**, **2610** may be adapted to communicate data, control, or management messages from the processor **2602** to applications installed on the host device and/or a remote device. As another example, one or

more of the interfaces **2606**, **2608**, **2610** may be adapted to allow a user or user device (e.g., personal computer (PC), etc.) to interact/communicate with the processing system **2600**. The processing system **2600** may include additional components not depicted in FIG. **26**, such as long term storage (e.g., non-volatile memory, etc.).

[0126] In some embodiments, the processing system **2600** is included in a network device that is accessing, or part otherwise of, a telecommunications network. In one example, the processing system **2600** is in a network-side device in a wireless or wireline telecommunications network, such as a base station, a relay station, a scheduler, a controller, a gateway, a router, an applications server, or any other device in the telecommunications network. In other embodiments, the processing system **2600** is in a user-side device accessing a wireless or wireline telecommunications network, such as a mobile station, a user equipment (UE), a personal computer (PC), a tablet, a wearable communications device (e.g., a smartwatch, etc.), or any other device adapted to access a telecommunications network.

[0127] In some embodiments, one or more of the interfaces **2606**, **2608**, **2610** connects the processing system **2600** to a transceiver adapted to transmit and receive signaling over the telecommunications network. FIG. **27** is a block diagram of a transceiver **2700** adapted to transmit and receive signaling over a telecommunications network. The transceiver **2700** may be installed in a host device. As shown, the transceiver **2700** comprises a network-side interface **2702**, a coupler **2704**, a transmitter **2706**, a receiver **2708**, a signal processor **2710**, and a device-side interface **2712**. The network-side interface **2702** may include any component or collection of components adapted to transmit or receive signaling over a wireless or wireline telecommunications network. The coupler **2704** may include any component or collection of components adapted to facilitate bi-directional communication over the network-side interface **2702**. The transmitter **2706** may include any component or collection of components (e.g., up-converter, power amplifier, etc.) adapted to convert a baseband signal into a modulated carrier signal suitable for transmission over the network-side interface **2702**. The receiver **2708** may include any component or collection of components (e.g., down-converter, low noise amplifier, etc.) adapted to convert a carrier signal received over the network-side interface **2702** into a baseband signal. The signal processor **2710** may include any component or collection of components adapted to convert a baseband signal into a data signal suitable for communication over the device-side interface(s) **2712**, or vice-versa. The device-side interface(s) **2712** may include any component or collection of components adapted to communicate data-signals between the signal processor **2710** and components within the host device (e.g., the processing system **2600**, local area network (LAN) ports, etc.).

[0128] The transceiver **2700** may transmit and receive signaling over any type of communications medium. In some embodiments, the transceiver **2700** transmits and receives signaling over a wireless medium. For example, the transceiver **2700** may be a wireless transceiver adapted to communicate in accordance with a wireless telecommunications protocol, such as a cellular protocol (e.g., long-term evolution (LTE), etc.), a wireless local area network (WLAN) protocol (e.g., Wi-Fi, etc.), or any other type of wireless protocol (e.g., Bluetooth, near field communication

(NFC), etc.). In such embodiments, the network-side interface **2702** comprises one or more antenna/radiating elements. For example, the network-side interface **2702** may include a single antenna, multiple separate antennas, or a multi-antenna array configured for multi-layer communication, e.g., single input multiple output (SIMO), multiple input single output (MISO), multiple input multiple output (MIMO), etc. In other embodiments, the transceiver **2700** transmits and receives signaling over a wireline medium, e.g., twisted-pair cable, coaxial cable, optical fiber, etc. Specific processing systems and/or transceivers may utilize all of the components shown, or only a subset of the components, and levels of integration may vary from device to device.

[0129] Although the description has been described in detail, it should be understood that various changes, substitutions and alterations can be made without departing from the spirit and scope of this disclosure as defined by the appended claims. Moreover, the scope of the disclosure is not intended to be limited to the particular embodiments described herein, as one of ordinary skill in the art will readily appreciate from this disclosure that processes, machines, manufacture, compositions of matter, means, methods, or steps, presently existing or later to be developed, may perform substantially the same function or achieve substantially the same result as the corresponding embodiments described herein. Accordingly, the appended claims are intended to include within their scope such processes, machines, manufacture, compositions of matter, means, methods, or steps.

What is claimed:

1. A wireless device comprising:

at least one processor; and

a non-transitory computer readable storage medium storing instructions that, when executed by the at least one processor, cause the wireless device to perform operations including:

transmitting in a 60 GHz frequency band a wireless packet, the wireless packet comprising a first header and a second header, the first header carrying a plurality of bits, a logical value of a subset of the plurality of bits in the first header indicating presence of the second header in the wireless packet, both of the first header and the second header being physical layer packet headers, the first header carrying a first parameter for an IEEE 802.11ad packet type, the second header carrying a second parameter for an IEEE 802.11ay packet type, the second parameter indicating a number of bonded channels, wherein channel bonding is not supported by the IEEE 802.11ad packet type.

2. The wireless device of claim 1, wherein the second parameter for the IEEE 802.11ay packet type in the second header further comprises a multiple input multiple output (MIMO) type, a MIMO order, or a combination thereof.

3. The wireless device of claim 1, wherein the wireless packet is a control packet, and wherein the plurality of bits includes 24 bits.

4. The wireless device of claim 1, wherein the wireless packet is one of a single carrier (SC) packet or an orthogonal frequency-division multiplexed (OFDM) packet, and wherein the plurality of bits includes 48 bits.

5. The wireless device of claim 1, wherein the wireless packet further comprises a payload and a training field.

6. The wireless device of claim 1, wherein the wireless packet further comprises a first short training field, a second short training field, a first channel estimation field, and a second channel estimation field.

7. A wireless device comprising:
at least one processor; and

a non-transitory computer readable storage medium storing instructions that, when executed by the at least one processor, cause the wireless device to perform operations including:

receiving in a 60 GHz band a wireless packet, the wireless packet comprising a first header and a second header, the first header carrying a plurality of bits, both of the first header and the second header being physical layer packet headers, the first header carrying a first parameter for an IEEE 802.11ad packet type, the second header carrying a second parameter for an IEEE 802.11ay packet type, the second parameter indicating a number of bonded channels, wherein channel bonding is not supported by the IEEE 802.11ad packet type; and
determining presence of the second header in the wireless packet according to a logical value of a subset of the plurality of bits in the first header.

8. The wireless device of claim 7, wherein the second parameter for the IEEE 802.11ay packet type in the second header further comprises a multiple input multiple output (MIMO) type, a MIMO order, or a combination thereof.

9. The wireless device of claim 7, wherein the wireless packet is a control packet, and wherein the plurality of bits includes 24 bits.

10. The wireless device of claim 7, wherein the wireless packet is one of a single carrier (SC) packet or an orthogonal frequency-division multiplexed (OFDM) packet, and wherein the plurality of bits includes 48 bits.

11. The wireless device of claim 7, wherein the wireless packet further comprises a payload and a training field.

12. The wireless device of claim 7, wherein the wireless packet further comprises a first short training field, a second short training field, a first channel estimation field, and a second channel estimation field.

13. A chipset, wherein the chipset is configured to:
output a wireless packet to a system bus for transmission on a 60 GHz frequency band, the wireless packet comprising a first header and a second header, the first header carrying a plurality of bits, a logical value of a subset of the plurality of bits in the first header indicating presence of the second header in the wireless packet, both of the first header and the second header being physical layer packet headers, the first header carrying a first parameter for an IEEE 802.11ad packet type, the second header carrying a second parameter for an IEEE 802.11ay packet type, the second parameter

indicating a number of bonded channels, wherein channel bonding is not supported by the IEEE 802.11ad packet type.

14. The chipset of claim 13, wherein the second parameter for the IEEE 802.11ay packet type in the second header further comprises a multiple input multiple output (MIMO) type, a MIMO order, or a combination thereof.

15. The chipset of claim 13, wherein the wireless packet is a control packet, and wherein the plurality of bits includes 24 bits.

16. The chipset of claim 13, wherein the wireless packet is one of a single carrier (SC) packet or an orthogonal frequency-division multiplexed (OFDM) packet, and wherein the plurality of bits includes 48 bits.

17. The chipset of claim 13, wherein the wireless packet further comprises a payload and a training field.

18. The chipset of claim 13, wherein the wireless packet further comprises a first short training field, a second short training field, a first channel estimation field, and a second channel estimation field.

19. A chipset, wherein the chipset is configured to:
read a wireless packet from a system bus, the wireless packet received on a 60 GHz band, the wireless packet comprising a first header and a second header, the first header carrying a plurality of bits, both of the first header and the second header being physical layer packet headers, the first header carrying a first parameter for an IEEE 802.11ad packet type, the second header carrying a second parameter for an IEEE 802.11ay packet type, the second parameter indicating a number of bonded channels, wherein channel bonding is not supported by the IEEE 802.11ad packet type; and
determine presence of the second header in the wireless packet according to a logical value of a subset of the plurality of bits in the first header.

20. The chipset of claim 19, wherein the second parameter for the IEEE 802.11ay packet type in the second header further comprises a multiple input multiple output (MIMO) type, a MIMO order, or a combination thereof.

21. The chipset of claim 19, wherein the wireless packet is a control packet, and wherein the plurality of bits includes 24 bits.

22. The chipset of claim 19, wherein the wireless packet is one of a single carrier (SC) packet or an orthogonal frequency-division multiplexed (OFDM) packet, and wherein the plurality of bits includes 48 bits.

23. The chipset of claim 19, wherein the wireless packet further comprises a payload and a training field.

24. The chipset of claim 19, wherein the wireless packet further comprises a first short training field, a second short training field, a first channel estimation field, and a second channel estimation field.

* * * * *