



(51) International Patent Classification:

H04W 72/12 (2009.01) H04W 84/12 (2009.01)

(21) International Application Number:

PCT/US2018/039308

(22) International Filing Date:

25 June 2018 (25.06.2018)

(25) Filing Language:

English

(26) Publication Language:

English

(71) Applicant: INTEL CORPORATION [US/US]; 2200 Mission College Boulevard, Santa Clara, California 95054 (US).

(72) Inventors: RASHID, Mohammad Mamun; 6058 NE Whitewood Dr., Hillsboro, Oregon 97124 (US). CAVALCANTI, Dave; POB 6525, Beaverton, Oregon 97007 (US). CARIOU, Laurent; 5048 SW Florida St., Portland, Oregon 97219 (US).

(74) Agent: ZIMMERMAN, Mark C.; Hanley, Flight & Zimmerman, LLC, 150 S. Wacker Drive, Suite 2200, Chicago, Illinois 60606 (US).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM,

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

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

Declarations under Rule 4.17:

- as to the identity of the inventor (Rule 4.17(i))
- as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))

(54) Title: METHODS AND APPARATUS TO FACILITATE SEMI-STATIC SCHEDULING AND/OR LOW-OVERHEAD ACKNOWLEDGEMENT PROTOCOLS IN A LOCAL AREA NETWORK

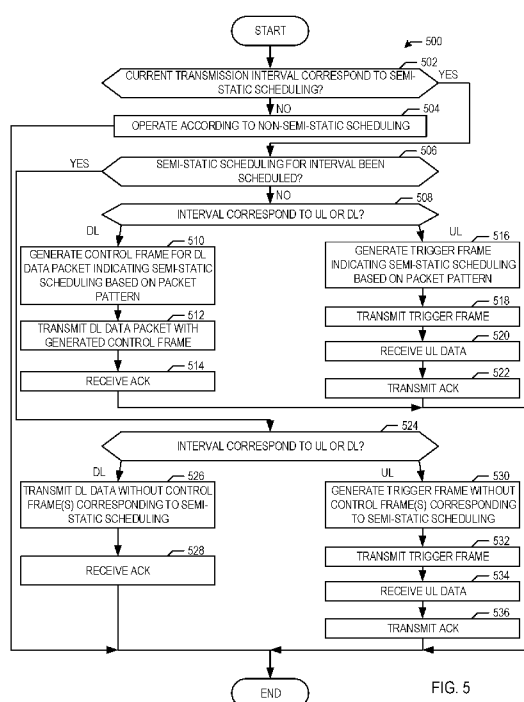


FIG. 5

(57) Abstract: Methods and apparatus to facilitate semi-static scheduling and/or low-overhead acknowledgement protocols in a wireless local area network are disclosed. An example apparatus includes a semi-static scheduler to determine if two or more transmission intervals correspond to a same transmission characteristic in a wireless local area network; and a packet generator to during a first transmission interval of the two or more transmission intervals, generate a first data packet including (A) a first value identifying when the two or more transmission intervals will occur and (B) a second value identifying the transmission characteristic.

Published:

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

METHODS AND APPARATUS TO FACILITATE SEMI-STATIC SCHEDULING AND/OR LOW-OVERHEAD ACKNOWLEDGEMENT PROTOCOLS IN A WIRELESS LOCAL AREA NETWORK

5 FIELD OF THE DISCLOSURE

This disclosure relates generally to wireless fidelity connectivity (Wi-Fi) and, more particularly, to methods and apparatus to facilitate semi-static scheduling and/or low-overhead acknowledgement protocols in a wireless local area network.

10 BACKGROUND

Many locations provide Wi-Fi to connect Wi-Fi enabled devices to networks such as the Internet. Wi-Fi enabled devices include personal computers, video-game consoles, mobile phones and devices, digital cameras, tablets, smart televisions, digital audio players, etc. Wi-Fi allows the Wi-Fi enabled devices to wirelessly access the Internet via a wireless local area network (WLAN). To provide Wi-Fi connectivity to a device, a Wi-Fi access point exchanges radio frequency Wi-Fi signals with the Wi-Fi enabled device within the access point (e.g., a hotspot) signal range. Wi-Fi is implemented using a set of media access control (MAC) and physical layer (PHY) specifications (e.g., such as the Institute of Electrical and Electronics Engineers (IEEE) 802.11 protocol).

20

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is an illustration of a communication system using wireless local area network Wi-Fi protocols to facilitate semi-static scheduling and/or acknowledgement protocols.

FIG. 2 is a block diagram of an example AP-based scheduling/ACK controller of FIG. 1.

25 FIG. 3 is a block diagram of an example STA-based scheduling/ACK controller of FIG. 1.

FIG. 4A and 4B is an illustration of an example synchronous transmission opportunity that includes frame/fields that may be generated by the example AP-based scheduling/ACK controller or the example STA-based scheduling/ACK controller of FIGS. 1-3.

FIG. 5 is a flowchart representative of example machine readable instructions that may be executed to implement the example AP controller of FIG. 2 based on a semi-static scheduling protocol.

FIGS. 6-7 are flowcharts representative of example machine readable instructions that may be executed to implement the example STA-based scheduling/ACK controller of FIG. 3 based on a semi-static scheduling protocol.

FIGS. 8-9 are flowcharts representative of example machine readable instructions that may be executed to implement the example AP-based scheduling/ACK controller of FIG. 2 based on an acknowledgement protocol.

FIGS. 10-11 are flowcharts representative of example machine readable instructions that may be executed to implement the example STA-based scheduling/ACK controller of FIG. 3 based on an acknowledgement protocol.

FIGS. 12A-H illustrate example acknowledgement protocol types.

FIG. 13 is a block diagram of a radio architecture in accordance with some examples.

FIG. 14 illustrates example front-end module circuitry for use in the radio architecture of FIG. 13 in accordance with some examples.

FIG. 15 illustrates example radio IC circuitry for use in the radio architecture of FIG. 13 in accordance with some examples.

FIG. 16 illustrates example baseband processing circuitry for use in the radio architecture of FIG. 13 in accordance with some examples.

FIG. 17 is a block diagram of a processor platform structured to execute the example machine readable instructions of FIGS. 5-11 to implement the example AP controller or the example STA controller of FIGS. 2 or 3.

The figures are not to scale. Wherever possible, the same reference numbers will be used throughout the drawing(s) and accompanying written description to refer to the same or like parts.

DETAILED DESCRIPTION

Various locations (e.g., homes, offices, coffee shops, restaurants, parks, airports, etc.) may provide Wi-Fi to the Wi-Fi enabled devices (e.g., stations (STA)) to connect the Wi-Fi enabled devices to the Internet, or any other network, with minimal hassle. The locations may provide one or more Wi-Fi access points (APs) to output Wi-Fi signals to the Wi-Fi enabled

devices within a range of the Wi-Fi signals (e.g., a hotspot). A Wi-Fi AP is structured to wirelessly connect a Wi-Fi enabled device to the Internet through a wireless local area network (WLAN) using Wi-Fi protocols (e.g., such as IEEE 802.11). The Wi-Fi protocol is the protocol specifying how the AP communicates with the devices to provide access to the Internet by transmitting uplink (UL) transmissions and receiving downlink (DL) transmissions to/from the Internet.

Some Wi-Fi networks have significant control overhead. Such significant overhead poses challenges in scaling to higher number of users and throughput. The overhead takes up significant airtime (bandwidth) causing inefficiency when an AP needs to serve (e.g., transmit data to) a STA multiple times. In such examples, the overhead may be redundant and avoidable. For example, multiple data transmission include the same or similar information in multiple headers. The larger the header, the more overhead in the transmission.

In some examples, synchronous transmission opportunity (S-TXOP) protocols may be used to reduce control overhead. S-TXOP includes generating a preamble including timing and frequency synchronization within each transmission interval of a transmission opportunity. Though the use of S-TXOP, each STA will know the time boundaries of each transmission interval, the number and duration of the intervals, etc. to maintain full synchronization prior to the transmission intervals to maintain full synchronization with the AP during the transmission opportunity, thereby decreasing the size of the preamble needed in the transmission intervals.

Examples disclosed herein further reduce control overhead to improve efficiency of next-generation Wi-Fi networks to support lower latency and higher capacity applications (e.g., autonomous systems, smart factories, professional audio/video and mobile/wireless VR are time-sensitive applications that require low and deterministic latency with high reliability). Examples disclosed herein utilize semi-static scheduling and/or low-overhead acknowledgments to reduce the amount of data needed in preambles of data packets during a transmission opportunity.

Additionally, many time-sensitive applications (e.g., VR, industrial, automation, etc.) have traffic characteristics that are periodic in nature. For example the traffic characteristics for a transmission interval used for packet transmission (e.g., UL or DL

transmission) may be repeated periodically within and/or across transmission opportunities. Such traffic/transmission characteristics include whether the transmission interval corresponds to UL or DL and/or resource unit allocation within the transmission interval.

Examples disclosed herein utilize semi-static scheduling for such periodic repetition of traffic characteristics to significantly reduce control overhead. Semi-static scheduling includes signaling a resource allocation for a transmission interval once and reusing the resource allocation information for subsequent transmission intervals corresponding to a periodic pattern. In this manner, the subsequent transmission intervals will have significantly reduced preambles, because the preambles do not need to include resource allocation information already provided in the initial transmission interval. Examples disclosed herein leverage the tightly synchronized nature of S-TXOP to perform such semi-static scheduling protocols. Using examples disclosed herein, throughput, latency, and capacity performance of Wi-Fi are improved.

Additionally, examples disclosed herein provide a low-overhead acknowledgement (ACK) signaling protocol within the S-TXOP framework to facilitate a low-overhead ACK. An ACK is a signal (e.g., a data packet or frame including data fields) that is transmitted from a device (e.g., a STA) that has received data packets from a sending device (e.g., AP or another STA) to verify that the data was received and/or not received. Examples disclosed herein provide a physical layer data packet (e.g., PHY protocol data unit (PPDU)) used for ACK signaling within a S-TXOP. The example ACK includes a lite-preamble with a small bitmap conveying ACK information, thereby corresponding to a shorter/smaller ACK than conventional ACK signals. Due to the information provided by the S-TXOP preamble, the disclosed ACK signal can provide limited information that can be inferred by a receiving device based on an ACK signaling protocol. Example ACK signaling protocols disclosed herein free up resources and improve efficiency and throughput performance of next-generation Wi-Fi networks. Additionally, by reducing the airtime spend sending an ACK, examples disclosed herein can support lower latency targets and/or high capacity for time-sensitive applications.

FIG. 1 illustrates an example communication system 100 using wireless local area network Wi-Fi protocols to facilitate semi-static scheduling and/or acknowledgement protocols. The example of FIG. 1 includes an example AP 102, example STAs 104, 106, 108, and an

example network 116. The example AP 102 includes example radio architecture 110 and the example AP-based scheduling/ACK controller 112. The example STAs 104, 106, 108 include the example radio architecture 110 and the example STA-based scheduling/ACK controller 114.

The example AP 102 of FIG. 1 is a device that allows the example STAs 104, 106, 108 to
5 wirelessly access the example network 116. The example AP 102 may be a router, a modem-router, and/or any other device that provides a wireless connection to the network 116. A router provides a wireless communication link to a STA. The router accesses the network 116 through a wire connection via a modem. A modem-router combines the functionalities of the modem and the router. In some examples, the AP 102 is a STA that is communication in the example STAs
10 104, 106, 108. The example radio architecture 110 of the AP 102 corresponds to components used to wirelessly transmit and/or receive data, as further described below in conjunction with FIG. 13. The example AP 102 includes the example AP-based scheduling/ACK controller 112 to facilitate a semi-static scheduling and/or acknowledgement protocols with the example STAs 104, 106, 108. Additionally, the example AP 102 may include an application processor (e.g., the
15 example application processor 1310 of FIG. 13) to generate instructions related to other Wi-Fi protocols.

The example STAs 104, 106, 108 of FIG. 1 is a Wi-Fi enabled computing devices. The example STAs 104, 106, 108 may be, for example, computing devices, portable devices, mobile devices, mobile telephones, smart phones, tablets, gaming systems, digital cameras, digital video
20 recorders, televisions, set top boxes, e-book readers, automated systems, VR-enabled devices, and/or any other Wi-Fi enabled devices. The example STAs 104, 106, 108 include the example STA-based scheduling/ACK controller 114 to facilitate a semi-static scheduling and/or acknowledgement protocols with the example AP 102. The example radio architecture 110 of the STAs 104, 106, 108 corresponds to components used to wirelessly transmit and/or receive
25 data, as further described below in conjunction with FIG. 13. Additionally, the example STAs 104, 106, 108 may include an application processor (e.g., the example application processor 1310 of FIG. 13) to generate instructions related to other Wi-Fi protocols.

The example AP-based scheduling/ACK controller 112 of FIG. 1 facilitates semi-static scheduling and/or ACK signaling with the example STAs 104, 106, 108. For example, the AP-
30 based scheduling/ACK controller 112 determines, based on the initial negotiations, which transmission interval within and/or across transmission opportunity(ies) have the same

transmission characteristics (e.g., UL vs. DL, resource allocations, etc.). Once determined, the AP-based scheduling/ACK controller 112 generates a control frame identifying when the transmission intervals with the same characteristics will occur. For example, the AP-based scheduling/ACK controller 112 may determine that a transmission interval has the same characteristics at transmission interval $M, M+X, M+2X$, etc., where M is the initial transmission interval, and X is the period of the repeated pattern. In another example, the AP-based scheduling/ACK controller 112 may determine that a transmission interval has the same characteristics at the same transmission interval of subsequent transmission opportunities. Accordingly, the AP-based scheduling/ACK controller 112 generates a control frame identifying the time intervals and/or transmission opportunities where the transmission characteristics will repeat. During a UL transmission, the AP-based scheduling/ACK controller 112 will include the control frame as part of a lite preamble for a trigger frame of the initial transmission interval (e.g., M) to identify the semi-static schedule of repeated transmission characteristics. During a DL transmission, the AP-based scheduling/ACK controller 112 will include the control frame as part of a lite preamble for the DL packet of the initial transmission interval (e.g., M) to identify the semi-static schedule of repeated transmission characteristics. During the subsequent transmission corresponding to the semi-static schedule ($M+X, M+2X$, etc. or in a subsequent transmission opportunity), the AP-based scheduling/ACK controller 112 removes/omits the control frame from the trigger frame/DL packet. In this manner, the receiving device (e.g., the example STAs 104, 106, 108) can process the initial control frame to determine the transmission characteristics during the initial data transmission interval and operate according the transmission characteristics during subsequent transmission intervals corresponding to the semi-static schedule without the AP 102 resending the control frame for each subsequent transmission interval, thereby increasing efficiency of data transmission corresponding to repeated transmission characteristics.

Additionally, the example AP-based scheduling/ACK controller 112 of FIG. 1 facilitates ACK signaling with the example STAs 104, 106, 108. The example AP-based scheduling/ACK controller 112 generates an ACK that is a PHY PPDU (e.g., as opposed to a full MAC frame). In this manner, the size of the ACK is substantially reduced. For example, the AP-based scheduling/ACK controller 112 selects an ACK protocol corresponding to an ACK type/configuration into a control frame (e.g., STXOP-SIGB frame) of a STXOP preamble for a

transmission opportunity. The ACK type corresponds to an immediate ACK (e.g., an ACK for every transmission interval) or a delayed ACK (e.g., an ACK after two or more transmission intervals). If the ACK type corresponds to a delayed ACK, the ACK type may correspond to a first type (e.g., type 1), a second type (e.g., type 2), or a third type (e.g., type 3). The ACK type
5 may be selected based on a protocol and/or user and/or manufacturer preferences. An example of an immediate ACK signaling is further described below in conjunction with FIGS. 12A and 12B.

During DL transmissions, the type 1 delayed ACK (D-ACK) corresponds to a protocol where each transmission interval corresponds to a DL data transmission for a single STA (e.g.,
10 AP 102 transmits to STA 104 at interval 1, AP 102 transmits to STA 106 at interval 2, and AP 102 transmits to STA 108 at interval 3) and the STAs 104, 106, 108 all transmit a D-ACK at the same time to the AP 102 on different resource units (RUs) (e.g., subchannels within a frequency band) based on the transmission interval corresponding to when DL packets were received. For example, because the STA 104 received DL packets at the first interval, the STA 104 will
15 transmit a D-ACK on a first RU corresponding to the first transmission interval. Likewise, the STA 106 will transmit a D-ACK on a second RU corresponding to the second transmission interval and the STA 108 will transmit a D-ACK on a third RU corresponding to the third transmission interval. The RU/transmission interval correspondence/link may be preset and/or may be included in the control frame of the preamble for the transmission opportunity. An
20 example of a type 1 D-ACK signaling for DL transmission is further described below in conjunction with FIGS. 12C.

During UL transmissions, the type 1 D-ACK corresponds to a protocol where each transmission interval corresponds to a UL data transmission from a single STA (e.g., STA 104 transmits to AP 102 at interval 1, STA 106 transmits to AP 102 at interval 2, and STA 108
25 transmits to AP 102 at interval 3) and the APs 102 transmits a D-ACK to the STAs 104, 106, 108, where the D-ACK includes ACK information elements (ACK-IEs) based on the transmission interval corresponding to when UL packets were received. An ACK-IE includes a bitmap corresponding to the UL data from a particular STA. For example, because the STA 104 transmitted UL packets at the first interval, the AP 102 will transmit a D-ACK with a first ACK-
30 IE corresponding to the UL data from the STA 104 at a first position corresponding to the first transmission interval. Likewise, the D-ACK will include a second ACK-IE corresponding to UL

data from the STA 106 at a second position corresponding to the second transmission interval and a third ACK-IE corresponding to UL data from the STA 108 at a third position corresponding to the third transmission interval. The ACK-IE position/transmission interval correspondence/link may be preset and/or may be included in the control frame of the preamble
5 for the transmission opportunity. An example of a type 1 D-ACK signaling for UL transmission is further described below in conjunction with FIGS. 12F.

The type 2 ACK (D-ACK) corresponds to a protocol where each transmission interval corresponds to a DL/UL transmissions to/from the example STAs 104, 106, 108 at different RUs within the same transmission interval. For example, the STA 104 may transmit/receive UL/DL
10 packets using a first RU for two or more time intervals, the STA 106 may transmit/receive UL/DL packets using a second RU for the two or more time intervals, and the STA 108 may transmit/receive UL/DL packets using a third RU for the two or more time intervals. In response to the transmission, the receiving device transmits an ACK using the RU corresponding to the RU where the data transmission was received. For example, if the AP 102 transmits first DL
15 data to the first STA 104 on a first RU, second DL data to the second STA 106 on a second RU, and third DL data to the third STA 108 on a third RU, the first STA 104 responds with an ACK on the first RU, the second STA 106 responds with an ACK on the second RU, and the third STA 108 response with an ACK on the third RU. Because the STAs 104, 106, 108 respond with ACKs on the channels where that the DL data were received on, the example AP-based
20 scheduling/ACK controller 112 can infer which ACKs correspond to each data transmission without including identification information in the ACKs, thereby reducing the amount of data needed in the ACKs. For example, if an AP 102 transmits a DL packet on a first RU and receives an ACK on the first RU, the example AP-based scheduling/ACK controller 112 infers that the received ACK corresponds to the transmitted DL packet because the ACK was received
25 on the first ACK. An example of a type 2 D-ACK signaling for UL/DL transmission is further described below in conjunction with FIGS. 12D and 12G.

The type 3 D-ACK corresponds to a protocol where each transmission interval corresponds to a DL/UL transmission to/from the example STAs 104, 106, 108 at different RUs within the same transmission, where the RUs used by each STA 104, 106, 108 changes during
30 each transmission interval. In the type 3 D-ACK signaling protocol, the example AP-based scheduling/ACK controller 112 reserves different RUs for each STA 104, 106, 108 to send

ACKs on. The example AP-based scheduling/ACK controller 112 identifies the reserved RU for each STA 104, 106, 108 in a control frame (e.g., STXOP-SIGB) of the STXOP preamble. In this manner, the STAs 104, 106, 108 each transmit ACKs on their corresponding RUs throughout the transmission opportunity. An example of a type 3 D-ACK signaling for UL/DL transmission is further described below in conjunction with FIGS. 12E and 12H.

The example STA-based scheduling/ACK controller 114 of FIG. 1 facilitates semi-static scheduling and/or ACK signaling with the example AP 102. For example, the STA-based scheduling/ACK controller 114 may, during a transmission opportunity, may process a received DL data packet and/or trigger frame from the example AP 102 to determine if a semi-static scheduling control frame is included in the preamble. If the semi-static scheduling control frame is included in the preamble, the example STA-based scheduling/ACK controller 114 determines the semi-static scheduling characteristics (e.g., the semi-static scheduling frequency and/or scope) and resource allocation information of the corresponding transmission intervals. In this manner, the example STA-based scheduling/ACK controller 114 will know the resource allocation information of all subsequent transmission intervals corresponding to the semi-static frequency and/or scope and the example AP 102 will remove the control frame (e.g., STXOP-SIG-D) from the lite preambles of the corresponding data transmissions (E.g., DL data or trigger frames), thereby increasing the efficiency. In some examples, during the subsequent transmission intervals corresponding to semi-static scheduling, the example STA-based scheduling/ACK controller 114 may listen to try to sense the control frame corresponding to the semi-static scheduling, in case the AP 102 updates the semi-static scheduling. In such examples, if the STA-based scheduling/ACK controller 114 senses the control frame, the example STA-based scheduling/ACK controller 114 updates the semi-static schedule consistent with the new control frame and if the STA-based scheduling/ACK controller 114 does not sense the control frame, the example STA-based scheduling/ACK controller 114 operates based on the previously received semi-static scheduling information and resource allocation.

Additionally, the example STA-based scheduling/ACK controller 114 of FIG. 1 facilitates ACK signaling with the example AP 102. For example, during an immediate ACK signaling protocol (e.g., or a type 2 D-ACK protocol) for DL data, the example STA-based scheduling/ACK controller 114 may generate an ACK including a bitmap corresponding to which DL data packets were received in a transmission interval at a particular RU. The example

STA-based scheduling/ACK controller 114 transmits the ACK on the RU where the data packets were received. In this manner, when the AP 102 receives multiple ACKs at different RUs from the different STAs 104, 106, 108, the AP 102 can infer which ACK belongs to which STA 104, 106, 108 based on the RU used to transmit the ACK. During an immediate ACK signaling
5 protocol (e.g., or a type 2 D-ACK protocol) for UL data, the example STA-based scheduling/ACK controller 114 receives an ACK at the RU used to transmit the UL data. In this manner, the example STA-based scheduling/ACK controller 114 can infer that the ACK corresponds to the UL data based on the RU of the ACK.

During a D-ACK signaling protocol, the example STA-based scheduling/ACK controller
10 114 of FIG. 1 responds with an ACK and/or infers which ACK corresponds to transmitted UL data based on the D-ACK protocol type (e.g., 1, 2, or 3). For example, in a type 1 protocol for DL data, if example STA-based scheduling/ACK controller 114 received DL data from the AP 102 during a first transmission interval, the example STA-based scheduling/ACK controller 114 transmits the ACK on an RU that corresponds to the first transmission interval based on a
15 interval-to-RU mapping identified in a control frame of a preamble for the transmission opportunity. In a type 1 protocol for UL data, if the example STA-based scheduling/ACK controller 114 transmits UL data to the AP 102 during a fourth transmission interval, the example STA-based scheduling/ACK controller 114 processes a received D-ACK from the AP 102 to determine the ACK-IE in a frame corresponding to the fourth transmission interval.
20 During a type 3 protocol for DL/UL data, the example STA-based scheduling/ACK controller 114 transmits ACKs to the AP 102 and/or infers which ACKs from the AP 102 correspond to transmitted UL data based on a predefined RU that selected for each STA 104, 106, 108 before the transmission opportunity and identified in the control frame of the preamble for the transmission opportunity.

25 The example network 116 of FIG. 1 is a system of interconnected systems exchanging data. The example network 116 may be implemented using any type of public or private network such as, but not limited to, the Internet, a telephone network, a local area network (LAN), a cable network, and/or a wireless network. To enable communication via the network 116, the example Wi-Fi AP 102 includes a communication interface that enables a connection to
30 an Ethernet, a digital subscriber line (DSL), a telephone line, a coaxial cable, or any wireless connection, etc.

FIG. 2 is a block diagram of an example implementation of the AP-based scheduling/ACK controller 112 of FIG. 1, disclosed herein, to facilitate semi-static scheduling and/or acknowledgement protocols. The example AP-based scheduling/ACK controller 112 includes an example component interface 200, an example interval tracker 202, an example
5 semi-static scheduler 204, an example packet generator 206, and an example ACK protocol processor 208.

The example component interface 200 of FIG. 2 interfaces with the application processor 1310 to transmit signals (e.g., instructions to operate according to a protocol) and/or receive signals (e.g., instructions corresponding to which ACK type to use) from the example application
10 processor 1310. Additionally, the example component interface 200 interfaces with the example the radio architecture 110 to instruct the radio architecture 110 to transmit data packet/frames and/or to receive data packets received from the radio architecture 110.

The example interval tracker 202 of FIG. 2 tracks the transmission intervals within a transmission opportunity. For example, for semi-static scheduling, the interval tracker 202
15 tracks the transmission intervals to determine if the current transmission interval corresponds to a semi-static schedule. For ACK protocoling, the interval tracker 202 tracks the intervals to identify when a D-ACK should be transmitted/received. Additionally, during a type 1 D-ACK, the interval tracker 302 may determine during which transmission interval each set of UL/DL data packets have been received in, in order to generate or interpret a type 1 D-ACK.

The example semi-static scheduler 204 of FIG. 2 schedules semi-static schedules for transmission intervals within or across transmission opportunities that have similar transmission characteristics (e.g., resource allocation information, whether the transmission opportunity
20 corresponds to UL or DL, etc.). The example semi-static scheduler 204 determines which transmission intervals correspond to such similar transmission characteristics and when the transmission intervals repeat in order to generate a semi-static schedule (e.g., corresponding to a frequency and scope). Accordingly, prior to the start of the transmission intervals within a transmission opportunity, the semi-static scheduler 204 determines which transmission
25 opportunities within and/or across transmission opportunities correspond to repeated transmission characteristics (e.g., UL vs. DL, resource allocations, etc.). The frequency of a semi-static schedule corresponds to how often the repeated pattern occurs and the scope
30

corresponds to whether the repeated pattern occurs within the same transmission opportunity or across different transmission opportunities.

The example packet generator 206 of FIG. 2 generates data packets and/or frames corresponding to semi-static scheduling and/or ACK signaling. For example, the packet generator 206 may generate control frames for a preamble of the transmission opportunity, preambles for data packets during transmission intervals, trigger frames, and/or ACK packets. The packet generator 206 generates an ACK that is a PHY PPDU (e.g., as opposed to a full MAC frame). In this manner, the size of the ACK is substantially reduced. Example packets and/or frames that the packet generator 206 may generate are described below in conjunction with FIGS. 4A-4B.

The example ACK protocol processor 208 of FIG. 2 facilitates an ACK signaling protocol based on a type (e.g., immediate ACK, delayed ACK, type 1 D-ACK, type 2 D-ACK, and/or type 2 D-ACK). The type may be preset or based on instructions from the example application processor 1310 of FIG. 13. The example ACK protocol processor 208 determines how to ACK received UL data packets and how to infer which DL data that received ACK data packets correspond to. The ACK signaling protocols are further described below in conjunction with FIGS. 12A-12H.

FIG. 3 is a block diagram of an example implementation of the STA-based scheduling/ACK controller 114 of FIG. 1, disclosed herein, to facilitate semi-static scheduling and/or acknowledgement protocols. The example STA-based scheduling/ACK controller 114 includes an example component interface 300, an example interval tracker 302, an example packet processor 304, an example packet generator 306, an example semi-static schedule database 308, and an example ACK protocol processor 310.

The example component interface 300 of FIG. 3 interfaces with the application processor 1310 to transmit signals (e.g., instructions to operate according to a protocol) and/or receive signals from the example application processor 1310. Additionally, the example component interface 300 interfaces with the example the radio architecture 110 to instruct the radio architecture 110 to transmit data packet/frames and/or to receive data packets received from the radio architecture 110.

The example interval tracker 302 of FIG. 3 tracks the transmission intervals within a transmission opportunity. For example, for semi-static scheduling, the interval tracker 302

tracks the transmission intervals to determine if the current transmission interval corresponds to a semi-static schedule. For ACK protocoling, the interval tracker 302 tracks the intervals to identify when a D-ACK should be transmitted/received. Additionally, during a type 1 D-ACK, the interval tracker 302 may determine during which transmission interval each set of UL/DL data packets have been received in, in order to generate or interpret a type 1 D-ACK.

The example packet processor 304 of FIG. 3 processes received data packets from the example AP 102 to facilitate a semi-static scheduling and/or an ACK signaling protocol. For example, during a semi-static scheduling, the packet processor 304 processes a received trigger frame and/or received DL data packets to identify if a semi-static scheduling information is included in a control frame of the trigger frame or the preamble of the DL data packet. If there is semi-static information included in the trigger frame and/or DL data packet, the packet processor 304 may store the semi-static information in the example semi-static scheduling database 308. In this manner, the packet processor 304 can facilitate the semi-static schedule in subsequent transmission intervals. Additionally, the packet processor 304 can determine ACK signaling information from control frames of a transmission opportunity preamble and/or a received ACK signal to infer what the ACK information corresponds to. In some examples, the packet processor 304 updates semi-static schedules in the semi-static schedule database 308 based on updated semi-static schedules in a received control frame.

The example packet generator 306 of FIG. 3 generates data packets and/or frames corresponding to semi-static scheduling and/or ACK signaling. For example, the packet generator 306 may generate ACK packets corresponding to the ACK signaling type currently being implemented. Example packets and/or frames that the packet generator 306 may generate are described below in conjunction with FIGS. 4A-4B.

The example ACK protocol processor 310 of FIG. 3 facilitates an ACK signaling protocol based on a type (e.g., immediate ACK, delayed ACK, type 1 D-ACK, type 3 D-ACK, and/or type 3 D-ACK). The type may be preset or based on instructions from the example application processor 1310 of FIG. 13. The example ACK protocol processor 310 determines how to ACK received DL data packets and how to infer which UL data that received ACK data packets correspond to. The ACK signaling protocols are further described below in conjunction with FIGS. 12A-12H.

FIGS. 4A-4B illustrate example field/frames that may be generated by the AP 102 and/or the STAs 104, 106, 108 within an example S-TXOP 400. FIGS. 4A-B include the example synchronous transmission opportunity (S-TXOP) 400 for UL/DL transmission between the example AP 102 and the example STAs 104, 106, 108. The example S-TXOP 400 includes an example S-TXOP preamble 402 and an example DL/UL intervals 404a-n. The example S-TXOP preamble 402 includes an example STXOP-SIG-A1 control field 408a, an example STXOP-SIG-A2 control field 408b, and an example STXOP-SIG-B control field 410. The example UL/DL interval 404a-n may correspond to the example DL interval 404a or the example UL interval 404b. The example DL interval 404a includes an example DL PPDU 412 and an example ACK 414a, b and the example UL interval 404b includes the example ACK 419a, b, an example lightweight (LW) trigger frame 416, an example UL lite preamble (LP) 417, and an example UL PPDU 418. The example DL PPDU 412 includes an example RSYNC field 420, an example STXOP-SIG-D field 421 including an example enhanced (E) HE-SIG-A field 422 and E-HE-SIG-B 423, an example HE-LTF frame(s) 424, and example data 426. The example light weight trigger frame 416 includes the example RSYNC field 420, an example semi-static scheduling frequency 428, and an example semi-static scheduling scope 430. The example ACK 414a includes an example UL lite preamble 444 and an example SIG-ACK field 446. The example ACK 419a includes an example DL lite preamble 450 and an example SIG-ACK 452. The example ACK 419b includes the example DL lite preamble 450, an example D-ACK configuration field 454, and an example SIG-DACK 456. The example ACK 414b includes the example UL lite preamble 444, and an example SIG-DACK 448. Although, the example S-TXOP 400 may include three intervals for UL/DL transmission, the S-TXOP 400 may include any number of intervals corresponding to any duration of time and/or any transmission type (e.g., UL or DL). Additionally, some field/frames may be rearranged, excluded, or added in the example of FIG. 4.

The example transmission opportunity 400 of FIG. 4A includes the example S-TXOP preamble 402 and a predetermined number of transmission intervals corresponding to either UL or DL transmission (e.g., DL/UL transmissions 404a-n). The example S-TXOP preamble 402 includes the example STXOP-SIG-A1 field 408a and the STXOP-SIG-A2 field 408b. The example STXOP-SIG-A1 field 408a and the STXOP-SIG-A2 field 408b of the example S-TXOP preamble 402 of FIG. 4 are control information fields including data corresponding to the timing

of the DL/UL transmission within the transmission opportunity. The STXOP-SIG-A1 field 408a and/or the STXOP-SIG-A2 field 408b include a number and duration of intervals inside the transmission opportunity after the S-TXOP preamble. For example, because the example S-TXOP 400 includes n UL/DL intervals, the example STXOP-SIG-A1 field 408a and/or the STXOP-SIG-A2 field 408b include data identifying the n intervals and the duration of each interval. If all of the control information data can be embedded into one of the example STXOP-SIG-A1 field 408a, the STXOP-SIG-A2 field 408b may repeat the data to increase the robustness of the data. Alternatively, if all of the control information data can be embedded into one of the example STXOP-SIG-A1 field 408a, the STXOP-SIG-A2 field 408b may be eliminated, thereby decreasing overhead.

The example STXOP-SIG-B field 410 of FIG. 4A is a control information field including data corresponding to ACK information. For example, the STXOP-SIG-B field 410 may include an ACK signaling to be used to DL transmission within the S-TXOP 400. The STXOP-SIG-B field 410 may include information related to whether the ACK signaling protocol is an immediate ACK protocol or a delayed ACK protocol. If the ACK is a D-ACK protocol, the example STXOP-SIG-B field 410 includes information corresponding a number of transmission intervals before an D-ACK is to be transmitted, the type of D-ACK (e.g., type 1, 2, or 3), and/or any D-ACK configuration information corresponding to the D-ACK type. For example, if the STXOP-SIG-B field 410 corresponds to a type 1 D-ACK, the D-ACK configuration information may correspond to a link between transmission intervals and RUs. In this manner, the STA may transmit an ACK on an RU corresponding to when the DL transmission was received (e.g., the transmission interval in which the DL transmission was received) and the AP 102 can infer which ACK corresponds to which DL packet based on the RU used to transmit the ACK. In another example, if the STXOP-SIG-B field 410 corresponds to a type 3 D-ACK, the D-ACK configuration information may correspond to a link between STAs and RUs (e.g., a first RU linked to the first STA 104, a second RU linked to the second STA 106, a third RU linked to the third STA 108). In this manner, the STA will always transmit an ACK on the linked RU throughout the S-TXOP 400 and the AP will infer which ACK corresponds to which STA based on the link.

The example DL interval 404a of FIG. 4A is reserved for DL transmission (e.g., from the AP 102 to the STA(s) 104, 106, 108). The example DL interval 404a includes the example DL

PPDU 412. The example DL PPDU is a portion of a MSDU that has been scrambled and encoded with a CRC by the example AP 102. The example AP 102 transmits the DL PPDU 412 during the DL interval 404a. In an immediate ACK protocol, once the DL PPDU 412 has been transmitted, the example STA(s) 104, 106, 108 may respond with the example ACK 414a, b
5 corresponding to which parts of the DL PPDU 412 has been received based on the ACK protocol. If the ACK protocol is a D-ACK protocol, the AP 102 continues to transmit additional DL PPDU 412 by transmitting a D-ACK, as further described below.

The example UL interval 404b of FIG. 4A is reserved for UL transmission (e.g., from the STAs 104, 106, 108 to the AP 102). The example UL interval 404b includes the LW trigger
10 PPDU 416 to initiate the UL transmission. The example LW trigger PPDU 416 is a control signal sent from the AP 102 to the STAs 104, 106, 108 to initiate UL transmission. In some examples, the LW trigger PPDU 416 identifies a semi-static schedule, as further explained below. The LW trigger PPDU 416 may correspond to spatial streams and/or orthogonal frequency division multiplexing (OFDMA) allocations for each connected STA and corresponds
15 to the exact moment when the example STAs 104, 106, 108 should initiate UL transmission. The LW trigger PPDU 416 includes the example RSYNC field 420 and the example STXOP-SIG-D field 421 including the example E-HE SIG-A 422 and the example E-HE-SIG-B 423, as further described below. The example LW trigger PPDU 416 is much shorter than a PPDU transporting a conventional trigger frame (e.g., a trigger frame including a full MAC layer
20 frame), because the LW trigger PPDU 416 does not include a MAC frame. Rather, the LW trigger PPDU 416 includes uplink resource allocation information in the example STXOP-SIG-D field 421 following the RSYNC field 420. The example UL interface 404b includes the example UL LP 419a, b, as further described below. Additionally, the example UL interval 404b includes the example UL PPDU 418. The example UL PPDU 418 is a portion of a MSDU that has been
25 scrambled and encoded with a CRC by the example STAs 104, 106, 108. The example STAs 104, 106, 108 transmit the UL PPDU 418 during the UL interval 404b. In an immediate ACK protocol, once the UL PPDU 418 has been transmitted, the example AP 102 responds with the example ACK 419a, b corresponding to parts of the UL PPDU 418 that have been received based on the ACK signaling used for each interval. In some examples, during a D-ACK
30 protocol, the STAs 104, 106, 108 continuing sending UL PPDU 418 until enough transmission intervals have passed to transmit the D-ACK.

The example DL PPDU 412 and the example UL PPDU 418 of FIG. 4A include the example lite preamble 417. As described above, because the example S-TXOP preamble 402 synchronizes the STAs 104, 106, 108 and the AP 102 for the duration of the example S-TXOP 400, the preamble of individual transmission intervals can be sized down to remove legacy fields (e.g., the L-STF, L-LTF, L-SIG, and RL-SIG frames). However, alternating between UL and DL requires switching of RX and TX front ends of the example radio architecture 110 of FIG. 11. Such switching may cause small CFO every UL/DL transmission. Accordingly, the example lite preamble 417 includes the example RSYNC field 420 to correct such small CFOs. The example RSYNC field 420 is an OFDM symbol sync field that includes two symbol repetitions (e.g., for 4.2 microseconds (us)) prepended by a cyclic prefix (0.8 us). The OFDM symbol may be generated by a 64 point inverse fast Fourier transform (IFFT) with 52-point frequency coefficients (e.g., subcarriers). For example, the RSYNC field 420 may include 52 subcarriers (e.g., 20 Megahertz) where the first 26 are repeated. In this manner, the receiving device can determine and correct the small CFO based on the repeated pattern. Alternatively, the RSYNC field 420 may include a sequence based on Zadoff-Chu or m-sequences.

The example lite preamble 417 of FIG. 4A further includes the example STXOP-SIG-D field 421. The example STXOP-SIG-D field 421 is a control information field (e.g., corresponding to UL transmission or DL transmission, depending on the data transmission type) including data related to resource allocation information for the example STAs 104, 106, 108. The example STXOP-SIG-D field 421 may be a combination of the HE-SIG-A field and the HE-SIG-B field with enhancements (e.g., the E-HE-SIG-A field 422 and the E-HE-SIG-B field 423) corresponding to semi-static scheduling. For example, when the DL PPDU 412 corresponds to a transmission interval that is to be semi-statically scheduled, the STXOP-SIG-D field 421 includes a semi-static scheduling frequency and a semi-static scheduling scope. The E-HE-SIG-A field 422 includes a value representative of whether a semi-static schedule is to be established or updated. If the value of the E-HE-SIG-A field 422 is non-zero, the value corresponds to the frequency of the semi-static schedule (e.g., the frequency of the transmission intervals where the transmission information (resource allocation, UL vs. DL, etc.) will be repeated). Additionally, the E-HE-SIG-A field 422 includes a value representative of the semi-static scheduling scope. The scope corresponds to whether the frequency applies within the STXOP 400 or across multiple STXOPs. For example, if the scope corresponds to within the STXOP 400 and the

frequency is n , the semi-static schedule will correspond to every n th transmission. If the scope corresponds to across multiple STXOPs, the value will correspond to a transmission interval within a subsequent S-TXOP where the transmission characteristics will be the same. Because the transmission information corresponds to subsequent transmission intervals, the semi-static scheduling information allows the AP 102 to remove the STXOP-SIG-D field 421 for subsequent DL PPDU 412 corresponding to the semi-static schedule. Additionally, the example lite preamble 417 includes the example HE-LTF fields 424. The HE-LTF fields 424 are control information fields corresponding to the multiple parameters for interpolation/smoothing during UL/DL transmission. Once the example lite preamble 417 is transmitted, the example data 426 is transmitted.

The example LW trigger frame 416 of FIG. 4A includes the RSYNC 420, as described above. Additionally, the LW trigger frame 416 includes a TF type subfield 427. The TF type subfield 427 includes a value (0-16) that corresponds to the type of trigger frame being transmitted. In some examples, one of the reserved values for trigger frame types may be allocated to semi-static scheduling triggers. In such examples, if the example TP type subfield 427 includes a value representative of a semi-static scheduling trigger, the example fields 428, 430 correspond to semi-static scheduling information. For example, the semi-static scheduling frequency field 428 corresponds to the frequency of the semi-static schedule and the semi-static scheduling scope field 430 corresponds to the scope of the semi-static schedule. In this manner, the AP 102 can schedule semi-static scheduling for transmission intervals that correspond to UL transmissions.

The example ACK 414a of FIG. 4B corresponds to an immediate ACK transmitted from the STAs 104, 106, 108 to the AP 102 in response to receiving DL data packets. The example UL lite preamble 444 may include the RSYNC field 420 which may be used to correct small CFOs. Additionally, the example ACK 414a includes the example SIG-ACK field 446. The example SIG-ACK field 446 includes a bitmap that corresponds to the received data packets that are stored in a buffer.

The example ACK 414b of FIG. 4B corresponds to a delayed ACK transmitted from the STAs 104, 106, 108 to the AP 102 in response to receiving DL data packets across two or more transmission intervals. The example UL lite preamble 444 may include the RSYNC field 420 which may be used to correct small CFOs. Additionally, the example ACK 414b includes the

example SIG-DACK field 448. The example SIG-DACK field 448 includes a bitmap that corresponds to the received data packets in the order they were sent in time across the transmission intervals being acknowledged that are stored in a buffer.

The example ACK 419a of FIG. 4B corresponds to an immediate ACK transmitted to the STAs 104, 106, 108 from the AP 102 in response to receiving UL data packets. The example DL lite preamble 450 may include the RSYNC field 420 which may be used to correct small CFOs. Additionally, the example ACK 419a includes the example SIG-ACK field 452. The example SIG-ACK field 452 includes a bitmap that corresponds to the received data packets that are stored in a buffer.

The example ACK 419b of FIG. 4B corresponds to a delayed ACK transmitted to the STAs 104, 106, 108 from the AP 102 in response to receiving UL data packets across two or more transmission intervals. The example DL lite preamble 450 may include the RSYNC field 420 which may be used to correct small CFOs. Additionally, the example ACK 419b includes the example D-ACK configuration field 454. The D-ACK configuration field 454 information corresponding to the ACK type (e.g., type 1, type 2, type 3), and/or other D-ACK configuration information according to the type. For example, if the ACK type corresponds to type 1, the D-ACK configuration information may include information corresponding to a link between transmission intervals and the size and number of ACK-IEs in the SIG-DACK field 456. In this manner, the STAs 104, 106, 108 can determine which ACK-IEs correspond to the UL data based on the links. In another example, when the D-ACK corresponds to type 3, the D-ACK configuration information may correspond to a link between STAs and RUs (e.g., a first RU linked to the first STA 104, a second RU linked to the second STA 106, a third RU linked to the third STA 108). In this manner, the STA will always transmit an ACK on the linked RU throughout the S-TXOP 400 and the AP will infer which ACK corresponds to which STA based on the link. Additionally, the example ACK 419b includes the example SIG-DACK field 456. The example SIG-DACK field 456 includes a bitmap that corresponds to the received data packets in the order they were sent in time across the transmission intervals being acknowledged that are stored in a buffer. The example ACKs 414a, 414b, 419a, 419b may be generated as PHY PPDU (e.g., as opposed to full MAC frames).

While an example manner of implementing the example AP-based scheduling/ACK controller 112 and/or the example STA-based scheduling/ACK controller 114 of FIG. 1 is

illustrated in FIGS. 2 and/or 3, one or more of the elements, processes and/or devices illustrated in FIGS. 2 and/or 3 may be combined, divided, re-arranged, omitted, eliminated and/or

implemented in any other way. Further, the example component interface 200, the example

interval tracker 202, the example semi-static scheduler 204, the example packet generator 206,

5 the example ACK protocol processor 208, the example component interface 300, the example

interval tracker 302, the example packet processor 304, the example packet generator 306, the

example semi-static schedule database 308, the example ACK protocol processor 310, and/or,

more generally the example AP-based scheduling/ACK controller 112 and/or the example STA-based scheduling/ACK controller 114 may be implemented by hardware, software, firmware

10 and/or any combination of hardware, software and/or firmware. Thus, for example, any of the

example component interface 200, the example interval tracker 202, the example semi-static

scheduler 204, the example packet generator 206, the example ACK protocol processor 208, the

example component interface 300, the example interval tracker 302, the example packet

processor 304, the example packet generator 306, the example semi-static schedule database 308,

15 the example ACK protocol processor 310, and/or, more generally the example AP-based

scheduling/ACK controller 112 and/or the example STA-based scheduling/ACK controller 114

could be implemented by one or more analog or digital circuit(s), logic circuits, programmable

processor(s), application specific integrated circuit(s) (ASIC(s)), programmable logic device(s)

(PLD(s)) and/or field programmable logic device(s) (FPLD(s)). When reading any of the

20 apparatus or system claims of this patent to cover a purely software and/or firmware

implementation, at least one of the example, the example component interface 200, the example

interval tracker 202, the example semi-static scheduler 204, the example packet generator 206,

the example ACK protocol processor 208, the example component interface 300, the example

interval tracker 302, the example packet processor 304, the example packet generator 306, the

25 example semi-static schedule database 308, the example ACK protocol processor 310, and/or,

more generally the example AP-based scheduling/ACK controller 112 and/or the example STA-

based scheduling/ACK controller 114 is/are hereby expressly defined to include a non-transitory

computer readable storage device or storage disk such as a memory, a digital versatile disk

(DVD), a compact disk (CD), a Blu-ray disk, etc., including the software and/or

30 firmware. Further still, the example AP-based scheduling/ACK controller 112 of FIG. 2 and/or

the example STA-based scheduling/ACK controller 114 of FIG. 3 may include one or more

elements, processes and/or devices in addition to, or instead of, those illustrated in FIG. 2 and/or 3, and/or may include more than one of any or all of the illustrated elements, processes and devices.

Flowcharts representative of example machine readable instructions for implementing the example AP-based scheduling/ACK controller 112 of FIG. 2 and/or the example STA-based scheduling/ACK controller 114 of FIG. 3 is shown in FIGS. 5-11. In this example, the machine readable instructions comprise a program for execution by a processor such as the processor 1712 shown in the example processor platform 1700 discussed below in connection with FIG. 17. The program may be embodied in software stored on a non-transitory computer readable storage medium such as a CD-ROM, a floppy disk, a hard drive, a digital versatile disk (DVD), a Blu-ray disk, or a memory associated with the processor 1712, but the entire program and/or parts thereof could alternatively be executed by a device other than the processor 1712 and/or embodied in firmware or dedicated hardware. Further, although the example program is described with reference to the flowchart illustrated in FIGS. 5-11, many other methods of implementing the example AP-based scheduling/ACK controller 112 and/or the example STA-based scheduling/ACK controller 114 may alternatively be used. For example, the order of execution of the blocks may be changed, and/or some of the blocks described may be changed, eliminated, or combined. Additionally or alternatively, any or all of the blocks may be implemented by one or more hardware circuits (e.g., discrete and/or integrated analog and/or digital circuitry, a Field Programmable Gate Array (FPGA), an Application Specific Integrated Circuit (ASIC), a comparator, an operational-amplifier (op-amp), a logic circuit, etc.) structured to perform the corresponding operation without executing software or firmware.

As mentioned above, the example processes of FIGS. 5-11 may be implemented using coded instructions (e.g., computer and/or machine readable instructions) stored on a non-transitory computer and/or machine readable medium such as a hard disk drive, a flash memory, a read-only memory, a compact disk, a digital versatile disk, a cache, a random-access memory and/or any other storage device or storage disk in which information is stored for any duration (e.g., for extended time periods, permanently, for brief instances, for temporarily buffering, and/or for caching of the information). As used herein, the term non-transitory computer readable medium is expressly defined to include any type of computer readable storage device and/or storage disk and to exclude propagating signals and to exclude transmission media.

“Including” and “comprising” (and all forms and tenses thereof) are used herein to be open ended terms. Thus, whenever a claim lists anything following any form of “include” or “comprise” (e.g., comprises, includes, comprising, including, etc.), it is to be understood that additional elements, terms, etc., may be present without falling outside the scope of the corresponding claim. As used herein, when the phrase “at least” is used as the transition term in a preamble of a claim, it is open-ended in the same manner as the term “comprising” and “including” are open ended.

FIG. 5 is an example flowchart 500 representative of example machine readable instructions that may be executed by the example AP-based scheduling/ACK controller 112 of FIGS. 1 and/or 2 within the example AP 102 of FIG. 1 to facilitate a synchronous transmission opportunity in a wireless local area network (e.g., Wi-Fi network). Although the example of FIG. 5 is described in conjunction with the example AP 102 in the network of FIG. 1, the instructions may be executed by any type of AP in any network.

At block 502, the example semi-static scheduler 204 and/or the interval tracker 202 determines if the current transmission interval corresponds to semi-static scheduling. For example, initially, the semi-static scheduler 204 processes the transmission characteristics (e.g., RU allocations, whether the interval corresponds to UL transmission or DL transmission, etc.) to determine if a subsequently scheduled transmission interval has the same transmission characteristics. Once a semi-static schedule has already been set (e.g., by the example AP-based scheduling/ACK controller 112), the interval tracker 202 determines if the current transmission interval corresponds to the semi-static frequency/scope set forth when the semi-static scheduling was established. If the example semi-static scheduler 204 determines that the current transmission interval does not correspond to semi-static scheduling (e.g., subsequent transmission intervals do not correspond to the same transmission characteristics) (block 502: NO), the component interface 200 transmits instructions to the application processor 1310 to operate according to non-semi static scheduling (block 504).

If the example semi-static scheduler 204 determines that the current transmission interval does correspond to semi-static scheduling (block 502: YES), the interval tracker 202 determines if semi-static scheduling for the current transmission interval has been scheduled (block 506). If the example semi-static scheduler 204 previously established semi-static scheduling at a frequency/scope, the example interval tracker 202 tracks the transmission intervals to determine

if a transmission interval corresponds to a previously established semi-static scheduling. If the example interval tracker 202 determines that a semi-static scheduling for the current transmission interval has been scheduled (block 506: YES), the process continues to block 524, as further described below. If the example interval tracker 202 determines that a semi-static scheduling for the current transmission interval has not been scheduled (block 506: NO), the example interval tracker 202 determines if the current transmission interval corresponds to UL transmission or DL transmission (block 508).

If the example interval tracker 202 determines that the current transmission interval corresponds to DL transmission (block 508: DL), the example packet generator 206 generates a control frame for the DL packet indicating semi-static scheduling based on the packet pattern (e.g., the frequency and/or scope of the repeated characteristics) (block 510). For example, the packet generator 206 generates the example STXOP-SIG-D field 421 of FIG. 4A indicating the semi-static scheduling frequency and/or the semi-static scheduling scope. At block 512, the example component interface 200 transmits instructions to the example radio architecture 110 to transmit the DL packet with the generated control frame. At block 514, the example component interface 200 receives (e.g., via the example radio architecture 110) an ACK from the corresponding STA.

If the example interval tracker 202 determines that the current transmission interval corresponds to UL transmission (block 508: UL), the example packet generator 206 generates a control frame for a trigger frame indicating semi-static scheduling based on the packet pattern (e.g., the frequency and/or scope of the repeated characteristics) (block 516). For example, the packet generator 206 generates the example LW trigger frame 416 of FIG. 4A indicating the semi-static scheduling frequency and/or the semi-static scheduling scope. At block 518, the example component interface 200 transmits instructions to the example radio architecture 110 to transmit the trigger frame. At block 520, the example component interface 200 receives (e.g., via the example radio architecture 110) UL data from the corresponding STA. At block 522, the example component interface 200 transmits instructions to the example radio architecture 110 to transmit an ACK.

If the example interval tracker 202 determines that a semi-static scheduling for the current transmission interval has been scheduled (block 506: YES), the example interval tracker 202 determines if the current transmission interval corresponds to UL transmission or DL

transmission (block 524). If the example interval tracker 202 determines that the current transmission interval corresponds to DL transmission (block 524: DL), the example packet generator 206 generates DL packet without (e.g., omitting) a control packet indicating semi-static scheduling based on the packet pattern (block 526), because the receiving device is already
5 aware of the information of the control frame and transmission characteristics based on a previously transmitted control frame. In some examples, if the example packet generator 206 may include the control packet indicating an updated semi-static scheduling. At block 528, the example component interface 200 receives (e.g., via the example radio architecture 110) an ACK from the corresponding STA.

10 If the example interval tracker 202 determines that the current transmission interval corresponds to UL transmission (block 524: UL), the example packet generator 206 generates a trigger frame without (e.g., omitting) a control frame indicating semi-static scheduling (block 530). In some examples, if the example packet generator 206 may include the control packet in the trigger frame indicating an updated semi-static scheduling. At block 532, the example
15 component interface 200 transmits instructions to the example radio architecture 110 to transmit the trigger frame. At block 534, the example component interface 200 receives (e.g., via the example radio architecture 110) UL data from the corresponding STA. At block 536, the example component interface 200 transmits instructions to the example radio architecture 110 to transmit an ACK.

20 FIG. 6 is an example flowchart 600 representative of example machine readable instructions that may be executed by the example STA-based scheduling/ACK controller 114 of FIGS. 1 and/or 3 within one of the example STA 104, 106, 108 of FIG. 1 to facilitate semi-static scheduling. Although the example of FIG. 6 is described in conjunction with one of the example STAs 104, 106, 108 in the network of FIG. 1, the instructions may be executed by any type of
25 STA in any network.

At block 602, the example component interface 300 determines if a trigger frame from the example AP 102 was received. For example, the component interface 300 may interface with the radio architecture 110 to determine if a trigger frame has been received. If the example component interface 300 determines that the trigger frame has been received (block 602: YES),
30 the process continues to the flowchart 700 of FIG. 7, as further described below in conjunction with FIG. 7. If the example component interface 300 determines that a trigger frame has not

been received (block 602: NO), the example interval tracker 302 determines if the current interval corresponds to semi-static scheduling (block 604). For example, when a semi-static schedule is established (e.g., via reception of a control frame identifying the semi-static schedule in a previously transmitted data packet and/or trigger frame), the details of the semi-static
5 schedule is stored in the example semi-static schedule database 308. Accordingly, the interval tracker 302 tracks the transmission interval which the semi-static schedule(s) stored in the example semi-static schedule database 308 to determine if the current transmission interval corresponds to a semi-static schedule.

If the example interval tracker 302 determines that the current transmission interval does
10 not correspond to semi-static scheduling (block 604: NO), the component interface 300 transmits instructions to the example application processor 1310 to operate according to non-semi static scheduling (block 606). If the example interval tracker 302 determines that the current transmission interval corresponds to semi-static scheduling (block 604: YES), the example component interface 300 listens for (e.g., attempt to sense) a control frame from the example AP
15 102 (block 608). As described above, the AP 102 may transmit the example control frame STXOP-SIG-D field 421 to establish (e.g., initiate) a semi-static schedule and/or to update a previously established semi-static schedule. Additionally, once established, the AP 102 may remove the control frame for all transmission intervals corresponding to the semi-static schedule.

If the example component interface 300 determines that the control frame was not
20 received (block 610: NO), the process continues to block 618, as further described below. If the example component interface 300 determines that the control frame was received (block 610: YES), the example packet processor 304 determines (e.g., if the control frame is an initiation of a semi-static schedule) or updates (e.g., if the control frame is an update to a previously established semi-static schedule) the semi-static scheduling based on the received control frame (block 612).
25 For example, the packet processor 304 determines that semi-static frequency and/or scope of the semi-static schedule and store the details in the example semi-static schedule database 308. In this manner, the example interval tracker 302 can determine when a subsequent transmission interval will corresponds to the semi-static schedule based on the information in the example semi-static schedule database 308. At block 614, the example component interface 300 receives
30 the DL data via the radio architecture 110 based on the RU allocations identified in the control frame. At block 616, the example component interface 300 transmits an ACK based on the

received data packets. At block 618, the example component interface 300 receives, via the radio architecture 110, DL data based on characteristics (e.g., RU resource allocations) corresponding to previously received semi-static scheduling data (e.g., the RU allocation corresponding to a semi-static schedule established and stored in the semi-static schedule database 308). At block 620, the example radio architecture 110 transmits an ACK
5 corresponding to the received DL data.

FIG. 7 is an example flowchart 700 representative of example machine readable instructions that may be executed by the example STA-based scheduling/ACK controller 114 of FIGS. 1 and/or 2 within the example STA 104, 106, 108 of FIG. 1 to facilitate to facilitate semi-static scheduling when a trigger frame is received (e.g., block 602: YES of FIG. 6). Although
10 the example of FIG. 7 is described in conjunction with one of the example STAs 104, 106, 108 in the network of FIG. 1, the instructions may be executed by any type of STA in any network.

At block 702, the example interval tracker 302 determines if the current transmission interval corresponds to semi-static scheduling. As described above, the example interval tracker
15 302 may process the semi-static schedules in the semi-static schedule database 308 to determine if the current transmission interval corresponds to semi-static scheduling. If the example interval tracker 302 determines that the current transmission interval corresponds to semi-static scheduling (block 702: NO), the example component interface 300 instructs the application processor 1310 to operate according to non-semi static scheduling (block 704). If the example
20 interval tracker 302 determines that the current transmission interval does not correspond to semi-static scheduling (block 702: YES), the example packet processor 304 processes the trigger frame to determine if the trigger frame includes semi-static scheduling information. A trigger frame may include semi-static scheduling information when it (A) corresponds to a newly established semi-static schedule or (B) corresponds to an updated to a previously established
25 semi-static schedule. The example packet processor 304 determines if the trigger frame includes semi-static scheduling information if the trigger frame includes a trigger type field value corresponding to a semi-static scheduling trigger frame.

If the example packet processor 304 determines that the trigger frame includes semi-static scheduling information (block 706: YES), the process continues to block 714, as further
30 described below. If the example packet processor 304 determines that the trigger frame does not include semi-static scheduling information (block 706: NO), the example packet processor 304

determines the characteristics (e.g., RU allocations) corresponding to the previously receive semi-static scheduling data (e.g. the RU allocations when the semi-static schedule was established) using the information stored in the semi-static schedule database 308 (block 708). At block 710, the example component interface 300 instructs the example radio architecture 110 to transmit UL data to the example AP 102 based on the characteristics. At block 712, the example component interface 300 receives an ACK via the radio architecture 110.

At block 714, the example packet processor 304 determines the semi-static schedule based on the trigger frame. For example, the packet processor 304 determines the frequency and/or scope of the semi-static schedule as well as the characteristics (e.g., RU allocation) of the semi-static schedule. In this manner, when a subsequent transmission interval corresponds to the semi-static schedule, the trigger frame can omit the characteristics information and the STA-based scheduling/ACK controller 114 can operate according to the stored semi-static schedule information/characteristics. At block 716, the example semi-static schedule database 308 stores or updates the semi-static scheduling information in the semi-static schedule database 308 based on the determined semi-static schedule. For example, if a semi-static schedule has already been established for the current transmission interval, the example semi-static schedule database 308 updates the semi-static scheduling information based on the trigger frame. If a semi-static schedule has not already been established for the current transmission interval, the example semi-static schedule database 308 stores the initial semi-static scheduling information. At block 718, the example component interface 300 instructs the radio architecture 110 to transmit UL data based on the characteristics identified in the trigger frame. At block 720, the example component interface 300 receives, via the radio architecture 110, an ACK and the process returns to FIG. 6 to end.

FIG. 8 is an example flowchart 800 representative of example machine readable instructions that may be executed by the example AP-based scheduling/ACK controller 112 of FIGS. 1 and/or 2 within the example AP 102 of FIG. 1 to facilitate an ACK signaling protocol for a UL data transmission. Although the example of FIG. 8 is described in conjunction with the example AP 102 in the network of FIG. 1, the instructions may be executed by any type of AP in any network.

At block 802, the example component interface 200 receives, via the radio architecture 110, uplink data from one or more of the example STAs 104, 106, 108. At block 804, the

example ACK protocol processor 208 determines if the currently used ACK protocol is an immediate ACK protocol or a D-ACK protocol. The example ACK protocol processor 208 may receive instructions from the example application processor 1310 (e.g., via the component interface 200) to operate using an ACK protocol or a D-ACK protocol (e.g., of any type). In some examples, the ACK protocol is determined based on user and/or manufacture preferences and/or is preset in the example AP 102. If the example ACK protocol processor 208 determines that the ACK protocol is an immediate ACK protocol (block 804: IMMEDIATE), the packet generator 206 generates an ACK including a bitmap (e.g., the example ACK 414a, b or ACK 419a, b of FIG. 4) based on the received data on the receive RU (block 806). At block 808, the example component interface 200 instructs the radio architecture 110 to transmit the ACK including the bitmap on the RU corresponding to where the UL data was received. For example, if the UL data was received on a first RU, the ACK is transmitted on the first RU.

If the example ACK protocol processor 208 determines that the ACK protocol is an delayed ACK protocol (block 804: DELAYED), the interval tracker 202 determines if it is time to send the D-ACK (block 810). As described above, the D-ACK may be sent after multiple transmission intervals. Accordingly, the interval tracker 202 tracks the transmission intervals to determine when it is time to transmit the D-ACK. If the example interval tracker 202 determines that it is not time to send the D-ACK (block 810: NO), the component interface 200 continues to receive additional UL data packets until it is time to send the D-ACK (block 812). If the example interval tracker 202 determines that it is time to send the D-ACK (block 810: YES), the example packet generator 206 generates a lite preamble and control frame corresponding to the ACK type (e.g., type 1, type 2, or type 3, as further described above in conjunction with FIG. 1) (block 814). The control frame (e.g., the D-ACK Config field 419 of FIG. 4B) includes information indicating the D-ACK type and other D-ACK configuration information corresponding to the type).

If the example ACK protocol processor 208 is operating with the type 1 D-ACK (block 816: TYPE 1), the example packet generator 206 generates a D-ACK data packet including the lite preamble, the control frame, and ACK-IEs in order corresponding to the order of received UL data packets (block 818). For example, the ACK-IEs may be in sequential order corresponding the sequential order of the received UL data packets (e.g., the first ACK-IE corresponds to the first received UL data packets, the second ACK-IE corresponds to the second

received UL data packets, etc.). At block 820, the example component interface 200 instructs the radio architecture 110 to transmit the example D-ACK data packets to the example STAs 104, 106, 108.

If the example ACK protocol processor 208 is operating with the type 2 D-ACK (block 816: TYPE 2), the example packet generator 206 generates a D-ACK data packet including the lite preamble, the control frame, and ACK-IEs corresponding to RU used to receive the UL data (block 822). For example, if first UL data was received at a first RU, the packet generator 206 generates an ACK for the first UL data to be transmitted on the first RU. At block 824, the example component interface 200 instructs the radio architecture 110 to transmit the example D-ACK data packets on the RU corresponding to the received UL data to the example STAs 104, 106, 108.

If the example ACK protocol processor 208 is operating with the type 3 D-ACK (block 816: TYPE 3), the example packet generator 206 generates a D-ACK data packet including the lite preamble, the control frame, and ACK-IEs with identifiers identifying the STA 104, 106, 108 that the ACK corresponds to (block 826). At block 828, the example component interface 200 instructs the radio architecture 110 to transmit the example D-ACK data packets on the RU corresponding to predefined RU for each of the STAs 104, 106, 108 that was defined in a control frame of a preamble for the transmission opportunity.

FIG. 9 is an example flowchart 900 representative of example machine readable instructions that may be executed by the example AP-based scheduling/ACK controller 112 of FIGS. 1 and/or 2 within the example AP 102 of FIG. 1 to facilitate an ACK signaling protocol for a DL data transmission. Although the example of FIG. 9 is described in conjunction with the example AP 102 in the network of FIG. 1, the instructions may be executed by any type of AP in any network.

At block 902, the example component interface 200 transmits a control frame in the synchronous transmission opportunity (S-TXOP) preamble including ACK information. For example, the example STXOP-SIG-B 410 may include the ACK type (e.g., immediate, delayed type 1, 2, or 3), and/or a number of DL intervals getting acknowledged in a D-ACK resource allocations to different STAs for transmitting ACKS. At block 904, the example component interface 200 instructs the radio architecture 110 to transmit DL packets to one or more of the example STAs 104, 106, 108 according to a predefined transmission protocol (e.g.,

corresponding to one or more of the ACK types). At block 906, the example ACK protocol processor 208 determines if the currently used ACK protocol is an immediate ACK protocol or a D-ACK protocol. The example ACK protocol processor 208 may receive instructions from the example application processor 1310 (e.g., via the component interface 200) to operate using an ACK protocol or a D-ACK protocol (e.g., of any type). In some examples, the ACK protocol is determined based on user and/or manufacture preferences and/or is preset in the example AP 102.

If the example ACK protocol processor 208 determines that the ACK protocol is an immediate ACK protocol (block 906: IMMEDIATE), the component interface 200 receives ACKs from the example STAs 104, 106, 108 at different RUs (block 908). At block 910, the example ACK protocol processor 208 infers which of the received ACKs corresponds to the transmitted DL data packets based on the RUs used to transmit the DL data packets. For example, if the example component interface 200 transmits DL data on a first RU, then the corresponding STA will response with an ACK on the first RU. Accordingly, the ACK protocol processor 208 can infer that when an ACK is received on a RU, that the ACK corresponds to the DL packet that was transmitted on the same RU.

If the example ACK protocol processor 208 determines that the ACK protocol is a delayed ACK protocol (block 906: DELAYED), the interval tracker 202 determines if it is time to receive the D-ACK (block 912). As described above, the D-ACK may be sent after multiple transmission intervals. Accordingly, the interval tracker 202 tracks the transmission intervals to determine when it is time to receive the D-ACK. If the example interval tracker 202 determines that it is not time to send the D-ACK (block 912: NO), the component interface 200 continues to transmit additional DL data packets according to the ACK protocol until it is time to receive the D-ACK (block 914). If the example interval tracker 202 determines that it is time to send the D-ACK (block 912: YES), the example component interface 200 receives D-ACKs from the STAs 104, 106, 108 (block 916).

If the example ACK protocol processor 208 is operating with the type 1 D-ACK (block 918: TYPE 1), the example ACK protocol processor 208 infers which ACK corresponds to each transmitted DL data packet based on the RUS used to receive the ACK and/or the DL transmission interval order (block 920). For example, if DL data received from the STA 104 at a first transmission interval may correspond to an ACK being received at a first RU. Accordingly,

the ACK protocol processor 208 may determine the RU where the ACK was received and infer that the ACK is based on the DL data that was received at a corresponding transmission interval.

If the example ACK protocol processor 208 is operating with the type 2 D-ACK (block 918: TYPE 2), the example ACK protocol processor 208 infers which of the received ACKs corresponds to the transmitted DL data packets based on the RUs used to transmit the DL data packets (block 922). For example, if the example component interface 200 transmits DL data on a first RU, then the corresponding STA will response with an ACK on the first RU.

Accordingly, the ACK protocol processor 208 can infer that when an ACK is received on a RU, that the ACK corresponds to the DL packet that was transmitted on the same RU.

If the example ACK protocol processor 208 is operating with the type 3 D-ACK (block 918: TYPE 3), the example ACK protocol processor 208 infers which of the received ACKs corresponds to the transmitted DL data packets based on information from a control frame (e.g., STXOP-SIG-B 410 of FIG. 4A) that identifies RUs reserved for each STA to transmit ACKs on (block 924). For example, the control frame may identify that STA 104 is to transmit ACKs on a first RU during the transmission opportunity. Accordingly, the ACK protocol processor 208 determines that any ACK received on the first RU corresponds to DL data to the STA 104.

FIG. 10 is an example flowchart 1000 representative of example machine readable instructions that may be executed by the example STA-based scheduling/ACK controller 114 of FIGS. 1 and/or 3 within one of the example STA 104, 106, 108 of FIG. 1 to facilitate semi-static scheduling. Although the example of FIG. 10 is described in conjunction with the example STAs 104, 106, 108 in the network of FIG. 1, the instructions may be executed by any type and/or number of STAs in any network.

At block 1002, the example component interface 300 receives the S-TXOP preamble 402 for the transmission opportunity via the radio architecture 110. At block 1004, the example packet processor 304 determines ACK information based on the preamble 402. For example, the packet processor 304 may process the STXOP-SIG-B frame 410 of the S-TXOP preamble 402 to determine the ACK or D-ACK type, a number of transmission intervals before a D-ACK is transmitted, D-ACK configuration information, and/or resource allocation to different STAs for transmitting ACKs. At block 1006, the example component interface 300 receives DL data from the example AP 102.

At block 1008, the example ACK protocol processor 310 determines if the currently used ACK protocol is an immediate ACK protocol or a D-ACK protocol. The example ACK protocol processor 310 determines whether the ACK protocol is an immediate ACK or a D-ACK based on the ACK information determined at block 1004. If the example ACK protocol processor 310 determines that the ACK protocol is an immediate ACK protocol (block 1008: IMMEDIATE), the packet generator 306 generates an ACK including a bitmap (e.g., the example ACK 414a, 414b, 419a, 419b of FIG. 4A) based on the received data on the receive RU (block 1010). At block 1012, the example component interface 300 instructs the radio architecture 110 to transmit the ACK including the bitmap on the RU corresponding to where the DL data was received. For example, if the DL data was received on a first RU, the ACK is transmitted on the first RU.

If the example ACK protocol processor 310 determines that the ACK protocol is a D-ACK protocol (block 1008: DELAYED), the interval tracker 302 determines if it is time to send the D-ACK (block 1014). As described above, the D-ACK may be sent after multiple transmission intervals. Accordingly, the interval tracker 302 tracks the transmission intervals to determine when it is time to transmit the D-ACK. If the example interval tracker 302 determines that it is not time to send the D-ACK (block 1014: NO), the component interface 300 continues to receive additional DL data packets until it is time to send the D-ACK (block 1016). If the example interval tracker 302 determines that it is time to send the D-ACK (block 1014: YES), the example packet generator 306 generates a D-ACK (e.g., the example D-ACK 414b of FIG. 4B) (block 1018). The example D-ACK includes a bitmap corresponding to received DL data packets.

If the example ACK protocol processor 310 is operating with the type 1 D-ACK (block 1020: TYPE 1), the example component interface 300 of FIG. 3 instructs the radio architecture 110 to transmit the D-ACK on the RU corresponding to the transmission interval where the DL packet was received (block 1022). For example, if the DL data was received at a first transmission interval, the component interface 300 transmits the D-ACK on a RU corresponding to the first transmission interval. The RU-transmission interval correspondence is identified as part of the D-ACK characteristics in the control frame (e.g., STXOP-SIG-B 410 of FIG. 4A) of the transmission opportunity preamble (e.g., the S-TXOP preamble 402 of FIG. 4A). If the example ACK protocol processor 208 is operating with the type 2 D-ACK (block 1020: TYPE

2), the example component interface 300 instructs the radio architecture 110 to transmit the D-ACK on the RU corresponding to the RU where the DL data was received (block 1024). For example, if the DL data was received at a first RU, the component interface 300 transmits the D-ACK using the first RU. If the example ACK protocol processor 208 is operating with the type 3
 5 D-ACK (block 1020: TYPE 3), the example component interface 300 instructs the radio architecture 110 to transmit the D-ACK on the RU identified in the S-TXOP preamble (block 1028).

FIG. 11 is an example flowchart 1100 representative of example machine readable instructions that may be executed by the example STA-based scheduling/ACK controller 114 of
 10 FIGS. 1 and/or 3 within one of the example STA 104, 106, 108 of FIG. 1 to facilitate semi-static scheduling. Although the example of FIG. 11 is described in conjunction with the example STAs 104, 106, 108 in the network of FIG. 1, the instructions may be executed by any type and/or number of STAs in any network.

At block 1102, the example component interface 300 transmits, via the radio architecture
 15 110, UL packets to the example AP 102 according to a predefined transmission protocol (e.g., corresponding to one or more of the ACK types). At block 1104, the example ACK protocol processor 310 determines if the currently used ACK protocol is an immediate ACK protocol or a D-ACK protocol. In some examples, the example ACK protocol processor 310 determine that the protocol is an immediate ACK protocol or a delayed ACK protocol based on information in a
 20 received control frame (e.g., the example STXOP SIG-B 410) of the S-TXOP preamble 402.

If the example ACK protocol processor 310 determines that the ACK protocol is an immediate ACK protocol (block 1104: IMMEDIATE), the component interface 300 receives ACKs from the AP 102 at different RUs (block 1106). At block 1108, the example ACK
 25 protocol processor 310 infers which of the received ACKs corresponds to the transmitted UL data packets based on the RU used to transmit the UL data packets. For example, if the example component interface 300 transmits UL data on a first RU, then the AP 102 will response with an ACK on the first RU. Accordingly, the ACK protocol processor 310 can infer that ACK corresponding to the RU used to transmit the UL data is the corresponding ACK.

If the example ACK protocol processor 310 determines that the ACK protocol is a
 30 delayed ACK protocol (block 1104: DELAYED), the interval tracker 302 determines if it is time to receive the D-ACK (block 1110). As described above, the D-ACK may be sent after multiple

transmission intervals. Accordingly, the interval tracker 302 tracks the transmission intervals to determine when it is time to receive the D-ACK. If the example interval tracker 302 determines that it is not time to send the D-ACK (block 1110: NO), the example STA-based scheduling/ACK controller 114 continues to operate according the ACK protocol until it is time to receive the D-ACK (block 1112). If the example interval tracker 302 determines that it is time to send the D-ACK (block 1110: YES), the example component interface 300 receives D-ACKs from the example AP 102 on one or more RUs (block 1114).

At block 1116, the example packet processor 304 determines if the received ACK(s) correspond to a type 1, a type 2, or a type 3 D-ACK. For example, the packet processor 304 processes the received ACK(s) and determines, based on the D-ACK Config field 454 of the D-ACK 419a what the D-ACK type is. If the example packet processor 304 determines that the received ACK(s) correspond to the type 1 D-ACK (block 1116: TYPE 1), the example ACK protocol processor 310 infers which ACK-IE corresponds to the transmitted UL data packet based on the transmission interval when the UL data was transmitted (block 1118). For example, if the UL data was transmitted at a first transmission interval, the ACK-IE will correspond to the first transmission interval. The ACK-IE/transmission interval correspondence may be identified in the D-ACK configuration frame 454 or the STXOP-SIG-B frame 410. If the example packet processor 304 determines that the received ACK(s) correspond to the type 2 D-ACK (block 1116: TYPE 2), the example ACK protocol processor 310 infers which ACK corresponds to each data packet based on the RU used to transmit the UL data packet (block 1120). For example, if the UL data packet(s) were transmitted using a first RU, the example ACK protocol processor 310 infers that the ACK received at the first RU is the ACK corresponding to the transmitted UL data. If the example packet processor 304 determines that the received ACK(s) correspond to the type 3 D-ACK (block 1116: TYPE 3), the example ACK protocol processor 310 infers which ACK corresponds to the transmitted UL data packet based on information from the control frame (e.g., the example STXOP-SIG-B frame 410 of FIG. 4A) in the S-TXOP preamble 402 (e.g., which is predefined by the AP 102) (block 1122).

FIGS. 12A-H illustrate timing diagrams for the ACK signaling protocols described herein. FIG. 12A illustrates an example timing diagram 1200 for immediate ACK for DL transmission. FIG. 12B illustrates an example timing diagram 1206 for immediate ACK for UL transmission. FIG. 12C illustrates an example timing diagram 1212 for a type 1 D-ACK for DL

transmission. FIG. 12D illustrates an example timing diagram 1218 for a type 2 D-ACK for DL transmission. FIG. 12E illustrates an example timing diagram 1224 for a type 3 D-ACK for DL transmission. FIG. 12F illustrates an example timing diagram 1230 for a type 1 D-ACK for UL transmission. FIG. 12G illustrates an example timing diagram 1236 for a type 2 D-ACK for UL transmission. FIG. 12H illustrates an example timing diagram 1244 for a type 3 D-ACK for UL transmission. Although, the example timing diagrams of FIG. 12A-H correspond to the example STAs 104, 106, 108 (e.g., the example STA x may correspond to STA 104, the example STA y may correspond to STA 106, and the example STA z correspond to the STA 108), the timing diagrams may be used in conjunction with any number and/or type of STAs.

FIG. 12A illustrates the example AP 102 transmitting example DL aggregated data packets 1202 (e.g., A-MPDUs) to the example STAs (e.g., STAs x, y, and z) using different RUs. For example, RU0 is used for DL transmission to the STA x, RU_{n-1} is used for DL transmission to STA y, etc. In response to the DL data packets transmission during the transmission interval, the STAs x, y, and z respond with example SIG-ACKs 1204. The SIG-ACKs include a lite preamble (e.g., an RSYNC frame) with a bitmap corresponding to the DL packets received on the corresponding RU. For example, the STA x transmits a SIG-ACK to the AP 102 on the RU0.

FIG. 12B illustrates the example STAs x, y, and z transmitting example UL aggregated data packets 1208 (e.g., A-MPDUs) to the example AP 102 using different RUs. For example, RU0 is used for DL transmission from the STA x, RU_{n-1} is used for DL transmission from STA y, etc. In response to the UL data packets transmission during the transmission interval, the AP 102 responds with example SIG-ACKs 1210 to each STA on each RU where the UL data was received. The SIG-ACKs include a lite preamble (e.g., an RSYNC frame) with a bitmap corresponding to the DL packets received on the corresponding RU. For example, the STA x transmits a SIG-ACK to the AP on the RU0.

FIG. 12C illustrates the example AP 102 transmitting DL data packets 1214 across the frequency band to each STAs x, y, and z at different transmission intervals. For example, the Ap 102 transmits DL data to the STA x at the first transmission interval (t_0), the AP 102 transmits DL data to the STA y at the second transmission interval (t_1), and the AP 102 transmits DL data to the STA z at the nth transmission interval (t_n). In response to receiving the DL data, the example STAs z, y, and z transmit the example SIG-ACKs 1216 each on a RU corresponding to

the transmission interval. For example, the STA x transmits an ACK corresponding to the first transmission interval (t_0) using the first RU0, the STA y transmits an ACK corresponding to the second transmission interval (t_1) using the first RU1, etc.

FIG. 12D illustrates the example AP 102 transmitting DL data packets (e.g., the example
 5 DL MU PPDU 1220a-n) to the STAs x, y, and z at various transmission intervals with the same RU for each transmission interval. For example, the STA x receives DL data from the AP 102 using RU0 across all transmission intervals, the STA y receives DL data from the AP 102 using RUn-1 across all transmissions, etc. In response to receiving the DL data, the example STAs x, y, and z transmit the example SIG-ACKs 1216 on the RU corresponding to the RU used for the
 10 DL transmissions. For example, the STAs x, y, and z transmit the ACKs 1222 corresponding to the DL packet transmitted on the different resource units. In this manner, AP 102 can infer that the ACK on the RU0 corresponds to the DL data packet, because the STA x transmitted the ACK on RU0.

FIG. 12E illustrates the example AP 102 transmitting DL data packets (e.g., the example
 15 DL MU PPDU 1226a-n) at various transmission intervals with the different RU for each transmission interval. For example, the STA x receives DL data from the AP 102 using RU0 across at the first transmission interval and using the RUn-1 at the second transmission interval, etc. In response to receiving the DL data, the example STAs x, y, and z transmit the example SIG-ACKs 1228 on the RU corresponding to a preset RU for the STA identified in a control
 20 frame of the S-TXOP preamble. For example, the STA x transmits an ACK corresponding to the DL packet transmitted on the RU0, because the control frame identifies the RU0 for the STA x.

FIG. 12F illustrates the example STAs x, y, and z transmitting UL data packets 1232a-n across the frequency band at different transmission intervals. For example, the STA x transmits DL data to the AP 102 at the first transmission interval (t_0), the STA y transmits DL data to the
 25 AP 102 at the second transmission interval (t_1), and the STA z transmits DL data to the AP 102 at the nth transmission interval (t_n). In response to receiving the DL data, the example AP 102 transmits the D-ACK including the example ACK-IE 1234a-n each corresponding to a transmission interval. For example, the first ACK-IE 1234a corresponds to the first transmission interval (t_0), the second ACK-IE 1234b corresponding to the second transmission interval (t_1),
 30 etc. In this manner, the STAs x, y, and z can infer which ACK-IE 1234a-n corresponds to the UL data from the corresponding STA x, y, and z.

FIG. 12G illustrates the example STAs x, y, z, transmitting UL data packets (e.g., the example UL MU PPDU 1238a-n) at various transmission intervals with the same RU for each transmission interval. For example, the STA x transmits UL data to the AP 102 using RU0 across all transmission intervals, the STA y transmits UL data to the AP 102 using RUn-1 across all transmissions. In response to receiving the UL data, the example AP 102 transmits the example SIG-ACKs 1240 on the RU corresponding to the RU used for the UL transmissions. For example, the AP 102 transmits multiple ACKs 1222 corresponding to the UL packet transmitted on the different resource units. In this manner, STA x can infer that the ACK on the RU0 corresponds to the UL data packet, because the STA x transmitted the UL packet on RU0.

FIG. 12H illustrates the example STAs x, y, z, transmitting UL data packets (e.g., the example UL MU PPDU 1242a-n) at various transmission intervals with the different RU for each transmission interval. For example, the STA x transmits UL data to the AP 102 using RU0 across at the first transmission interval and using the RUn-1 at the second transmission interval, etc. In response to receiving the UL data, the example AP 102 transmits the example SIG-ACKs 1244 on the RU corresponding to a preset RU for the STA identified in a control frame of the S-TXOP preamble. For example, the AP 102 transmits an ACK corresponding to the UL packet transmitted on the RU0 (e.g., the UL packets to STA x), because the control frame identifies the RU0 for the STA x.

FIG. 13 is a block diagram of a radio architecture 110 in accordance with some embodiments that may be implemented in the example AP 102 and/or the example STAs 104, 106, 108 of FIG. 1. Radio architecture 110 may include radio front-end module (FEM) circuitry 1304a-b, radio IC circuitry 1306a-b and baseband processing circuitry 1308a-b. Radio architecture 110 as shown includes both Wireless Local Area Network (WLAN) functionality and Bluetooth (BT) functionality although embodiments are not so limited. In this disclosure, “WLAN” and “Wi-Fi” are used interchangeably.

FEM circuitry 1304a-b may include a WLAN or Wi-Fi FEM circuitry 1304a and a Bluetooth (BT) FEM circuitry 1304b. The WLAN FEM circuitry 1304a may include a receive signal path comprising circuitry configured to operate on WLAN RF signals received from one or more antennas 1301, to amplify the received signals and to provide the amplified versions of the received signals to the WLAN radio IC circuitry 1306a for further processing. The BT FEM circuitry 1304b may include a receive signal path which may include circuitry configured to

operate on BT RF signals received from one or more antennas 1301, to amplify the received signals and to provide the amplified versions of the received signals to the BT radio IC circuitry 1306b for further processing. FEM circuitry 1304a may also include a transmit signal path which may include circuitry configured to amplify WLAN signals provided by the radio IC circuitry 1306a for wireless transmission by one or more of the antennas 1301. In addition, FEM circuitry 1304b may also include a transmit signal path which may include circuitry configured to amplify BT signals provided by the radio IC circuitry 1306b for wireless transmission by the one or more antennas. In the embodiment of FIG. 13, although FEM 1304a and FEM 1304b are shown as being distinct from one another, embodiments are not so limited, and include within their scope the use of an FEM (not shown) that includes a transmit path and/or a receive path for both WLAN and BT signals, or the use of one or more FEM circuitries where at least some of the FEM circuitries share transmit and/or receive signal paths for both WLAN and BT signals.

Radio IC circuitry 1306a-b as shown may include WLAN radio IC circuitry 1306a and BT radio IC circuitry 1306b. The WLAN radio IC circuitry 1306a may include a receive signal path which may include circuitry to down-convert WLAN RF signals received from the FEM circuitry 1304a and provide baseband signals to WLAN baseband processing circuitry 1308a. BT radio IC circuitry 1306b may in turn include a receive signal path which may include circuitry to down-convert BT RF signals received from the FEM circuitry 1304b and provide baseband signals to BT baseband processing circuitry 1308b. WLAN radio IC circuitry 1306a may also include a transmit signal path which may include circuitry to up-convert WLAN baseband signals provided by the WLAN baseband processing circuitry 1308a and provide WLAN RF output signals to the FEM circuitry 1304a for subsequent wireless transmission by the one or more antennas 1301. BT radio IC circuitry 1306b may also include a transmit signal path which may include circuitry to up-convert BT baseband signals provided by the BT baseband processing circuitry 1308b and provide BT RF output signals to the FEM circuitry 1304b for subsequent wireless transmission by the one or more antennas 1301. In the embodiment of FIG. 13, although radio IC circuitries 1306a and 1306b are shown as being distinct from one another, embodiments are not so limited, and include within their scope the use of a radio IC circuitry (not shown) that includes a transmit signal path and/or a receive signal path for both WLAN and BT signals, or the use of one or more radio IC circuitries where at least

some of the radio IC circuitries share transmit and/or receive signal paths for both WLAN and BT signals.

Baseband processing circuitry 1308a-b may include a WLAN baseband processing circuitry 1308a and a BT baseband processing circuitry 1308b. The WLAN baseband processing circuitry 1308a may include a memory, such as, for example, a set of RAM arrays in a Fast Fourier Transform or Inverse Fast Fourier Transform block (not shown) of the WLAN baseband processing circuitry 1308a. Each of the WLAN baseband circuitry 1308a and the BT baseband circuitry 1308b may further include one or more processors and control logic to process the signals received from the corresponding WLAN or BT receive signal path of the radio IC circuitry 1306a-b, and to also generate corresponding WLAN or BT baseband signals for the transmit signal path of the radio IC circuitry 1306a-b. Each of the baseband processing circuitries 1308a and 1308b may further include physical layer (PHY) and medium access control layer (MAC) circuitry, and may further interface with the example AP-based scheduling/ACK controller 112 and/or the STA-based scheduling/ACK controller 114 (e.g., depending on which device the radio architecture 110 is implemented in) for generation and processing of the baseband signals and for controlling operations of the radio IC circuitry 1306a-b.

Referring still to FIG. 13, according to the shown embodiment, WLAN-BT coexistence circuitry 1313 may include logic providing an interface between the WLAN baseband circuitry 1308a and the BT baseband circuitry 1308b to enable use cases requiring WLAN and BT coexistence. In addition, a switch 1303 may be provided between the WLAN FEM circuitry 1304a and the BT FEM circuitry 1304b to allow switching between the WLAN and BT radios according to application needs. In addition, although the antennas 1301 are depicted as being respectively connected to the WLAN FEM circuitry 1304a and the BT FEM circuitry 1304b, embodiments include within their scope the sharing of one or more antennas as between the WLAN and BT FEMs, or the provision of more than one antenna connected to each of FEM 1304a or 1304b.

In some embodiments, the front-end module circuitry 1304a-b, the radio IC circuitry 1306a-b, and baseband processing circuitry 1308a-b may be provided on a single radio card, such as wireless radio card 1302. In some other embodiments, the one or more antennas 1301, the FEM circuitry 1304a-b and the radio IC circuitry 1306a-b may be provided on a single radio

card. In some other embodiments, the radio IC circuitry 1306a-b and the baseband processing circuitry 1308a-b may be provided on a single chip or integrated circuit (IC), such as IC 1312.

In some embodiments, the wireless radio card 1302 may include a WLAN radio card and may be configured for Wi-Fi communications, although the scope of the embodiments is not limited in this respect. In some of these embodiments, the radio architecture 110 may be configured to receive and transmit orthogonal frequency division multiplexed (OFDM) or orthogonal frequency division multiple access (OFDMA) communication signals over a multicarrier communication channel. The OFDM or OFDMA signals may comprise a plurality of orthogonal subcarriers.

In some of these multicarrier embodiments, radio architecture 110 may be part of a Wi-Fi communication station (STA) such as a wireless access point (AP), a base station or a mobile device including a Wi-Fi device. In some of these embodiments, radio architecture 110 may be configured to transmit and receive signals in accordance with specific communication standards and/or protocols, such as any of the Institute of Electrical and Electronics Engineers (IEEE) standards including, 802.11n-2009, IEEE 802.11-2012, IEEE 802.11-2016, 802.11n-2009, 802.11ac, 802.11ah, 802.11ad, 802.11ay and/or 802.11ax standards and/or proposed specifications for WLANs, although the scope of embodiments is not limited in this respect. Radio architecture 110 may also be suitable to transmit and/or receive communications in accordance with other techniques and standards.

In some embodiments, the radio architecture 110 may be configured for high-efficiency Wi-Fi (HEW) communications in accordance with the IEEE 802.11ax standard. In these embodiments, the radio architecture 110 may be configured to communicate in accordance with an OFDMA technique, although the scope of the embodiments is not limited in this respect.

In some other embodiments, the radio architecture 110 may be configured to transmit and receive signals transmitted using one or more other modulation techniques such as spread spectrum modulation (e.g., direct sequence code division multiple access (DS-CDMA) and/or frequency hopping code division multiple access (FH-CDMA)), time-division multiplexing (TDM) modulation, and/or frequency-division multiplexing (FDM) modulation, although the scope of the embodiments is not limited in this respect.

In some embodiments, as further shown in FIG. 13, the BT baseband circuitry 1308b may be compliant with a Bluetooth (BT) connectivity standard such as Bluetooth, Bluetooth 4.0 or

Bluetooth 10.0, or any other iteration of the Bluetooth Standard. In embodiments that include BT functionality as shown for example in FIG. 13, the radio architecture 110 may be configured to establish a BT synchronous connection oriented (SCO) link and or a BT low energy (BT LE) link. In some of the embodiments that include functionality, the radio architecture 110 may be configured to establish an extended SCO (eSCO) link for BT communications, although the scope of the embodiments is not limited in this respect. In some of these embodiments that include a BT functionality, the radio architecture may be configured to engage in a BT Asynchronous Connection-Less (ACL) communications, although the scope of the embodiments is not limited in this respect. In some embodiments, as shown in FIG. 13, the functions of a BT radio card and WLAN radio card may be combined on a single wireless radio card, such as single wireless radio card 1302, although embodiments are not so limited, and include within their scope discrete WLAN and BT radio cards

In some embodiments, the radio-architecture 110 may include other radio cards, such as a cellular radio card configured for cellular (e.g., 5GPP such as LTE, LTE-Advanced or 7G communications).

In some IEEE 802.11 embodiments, the radio architecture 110 may be configured for communication over various channel bandwidths including bandwidths having center frequencies of about 900 MHz, 2.4 GHz, 5 GHz, and bandwidths of about 2 MHz, 4 MHz, 5 MHz, 5.5 MHz, 6 MHz, 8 MHz, 10 MHz, 20 MHz, 40 MHz, 80 MHz (with contiguous bandwidths) or 80+80 MHz (160MHz) (with non-contiguous bandwidths). In some embodiments, a 920 MHz channel bandwidth may be used. The scope of the embodiments is not limited with respect to the above center frequencies however.

FIG. 14 illustrates WLAN FEM circuitry 1304a in accordance with some embodiments. Although the example of FIG. 14 is described in conjunction with the WLAN FEM circuitry 1304a, the example of FIG. 14 may be described in conjunction with the example BT FEM circuitry 1304b (FIG. 13), although other circuitry configurations may also be suitable.

In some embodiments, the FEM circuitry 1304a may include a TX/RX switch 1402 to switch between transmit mode and receive mode operation. The FEM circuitry 1304a may include a receive signal path and a transmit signal path. The receive signal path of the FEM circuitry 1304a may include a low-noise amplifier (LNA) 1406 to amplify received RF signals 1403 and provide the amplified received RF signals 1407 as an output (e.g., to the radio IC

circuitry 1306a-b (FIG. 13)). The transmit signal path of the circuitry 1304a may include a power amplifier (PA) to amplify input RF signals 1409 (e.g., provided by the radio IC circuitry 1306a-b), and one or more filters 1412, such as band-pass filters (BPFs), low-pass filters (LPFs) or other types of filters, to generate RF signals 1415 for subsequent transmission (e.g., by one or more of the antennas 1301 (FIG. 13)) via an example duplexer 1414.

In some dual-mode embodiments for Wi-Fi communication, the FEM circuitry 1304a may be configured to operate in either the 2.4 GHz frequency spectrum or the 5 GHz frequency spectrum. In these embodiments, the receive signal path of the FEM circuitry 1304a may include a receive signal path duplexer 1404 to separate the signals from each spectrum as well as provide a separate LNA 1406 for each spectrum as shown. In these embodiments, the transmit signal path of the FEM circuitry 1304a may also include a power amplifier 1410 and a filter 1412, such as a BPF, a LPF or another type of filter for each frequency spectrum and a transmit signal path duplexer 1404 to provide the signals of one of the different spectrums onto a single transmit path for subsequent transmission by the one or more of the antennas 1301 (FIG. 13). In some embodiments, BT communications may utilize the 2.4 GHz signal paths and may utilize the same FEM circuitry 1304a as the one used for WLAN communications.

FIG. 15 illustrates radio IC circuitry 1306a in accordance with some embodiments. The radio IC circuitry 1306a is one example of circuitry that may be suitable for use as the WLAN or BT radio IC circuitry 1306a/1306b (FIG. 13), although other circuitry configurations may also be suitable. Alternatively, the example of FIG. 15 may be described in conjunction with the example BT radio IC circuitry 1306b.

In some embodiments, the radio IC circuitry 1306a may include a receive signal path and a transmit signal path. The receive signal path of the radio IC circuitry 1306a may include at least mixer circuitry 1502, such as, for example, down-conversion mixer circuitry, amplifier circuitry 1506 and filter circuitry 1508. The transmit signal path of the radio IC circuitry 1306a may include at least filter circuitry 1512 and mixer circuitry 1514, such as, for example, up-conversion mixer circuitry. Radio IC circuitry 1306a may also include synthesizer circuitry 1504 for synthesizing a frequency 1505 for use by the mixer circuitry 1502 and the mixer circuitry 1514. The mixer circuitry 1502 and/or 1514 may each, according to some embodiments, be configured to provide direct conversion functionality. The latter type of circuitry presents a much simpler architecture as compared with standard super-heterodyne mixer circuitries, and

any flicker noise brought about by the same may be alleviated for example through the use of OFDM modulation. FIG. 15 illustrates only a simplified version of a radio IC circuitry, and may include, although not shown, embodiments where each of the depicted circuitries may include more than one component. For instance, mixer circuitry 1514 may each include one or more
5 mixers, and filter circuitries 1508 and/or 1512 may each include one or more filters, such as one or more BPFs and/or LPFs according to application needs. For example, when mixer circuitries are of the direct-conversion type, they may each include two or more mixers.

In some embodiments, mixer circuitry 1502 may be configured to down-convert RF signals 1407 received from the FEM circuitry 1304a-b (FIG. 13) based on the synthesized
10 frequency 1505 provided by synthesizer circuitry 1504. The amplifier circuitry 1506 may be configured to amplify the down-converted signals and the filter circuitry 1508 may include a LPF configured to remove unwanted signals from the down-converted signals to generate output baseband signals 1507. Output baseband signals 1507 may be provided to the baseband processing circuitry 1308a-b (FIG. 13) for further processing. In some embodiments, the output
15 baseband signals 1507 may be zero-frequency baseband signals, although this is not a requirement. In some embodiments, mixer circuitry 1502 may comprise passive mixers, although the scope of the embodiments is not limited in this respect.

In some embodiments, the mixer circuitry 1514 may be configured to up-convert input baseband signals 1511 based on the synthesized frequency 1505 provided by the synthesizer
20 circuitry 1504 to generate RF output signals 1409 for the FEM circuitry 1304a-b. The baseband signals 1511 may be provided by the baseband processing circuitry 1308a-b and may be filtered by filter circuitry 1512. The filter circuitry 1512 may include a LPF or a BPF, although the scope of the embodiments is not limited in this respect.

In some embodiments, the mixer circuitry 1502 and the mixer circuitry 1514 may each
25 include two or more mixers and may be arranged for quadrature down-conversion and/or up-conversion respectively with the help of synthesizer 1504. In some embodiments, the mixer circuitry 1502 and the mixer circuitry 1514 may each include two or more mixers each configured for image rejection (e.g., Hartley image rejection). In some embodiments, the mixer circuitry 1502 and the mixer circuitry 1514 may be arranged for direct down-conversion and/or
30 direct up-conversion, respectively. In some embodiments, the mixer circuitry 1502 and the mixer

circuitry 1514 may be configured for super-heterodyne operation, although this is not a requirement.

Mixer circuitry 1502 may comprise, according to one embodiment: quadrature passive mixers (e.g., for the in-phase (I) and quadrature phase (Q) paths). In such an embodiment, RF input signal 1407 from FIG. 15 may be down-converted to provide I and Q baseband output signals to be sent to the baseband processor

Quadrature passive mixers may be driven by zero and ninety-degree time-varying LO switching signals provided by a quadrature circuitry which may be configured to receive a LO frequency (fLO) from a local oscillator or a synthesizer, such as LO frequency 1505 of synthesizer 1504 (FIG. 15). In some embodiments, the LO frequency may be the carrier frequency, while in other embodiments, the LO frequency may be a fraction of the carrier frequency (e.g., one-half the carrier frequency, one-third the carrier frequency). In some embodiments, the zero and ninety-degree time-varying switching signals may be generated by the synthesizer, although the scope of the embodiments is not limited in this respect.

In some embodiments, the LO signals may differ in duty cycle (the percentage of one period in which the LO signal is high) and/or offset (the difference between start points of the period). In some embodiments, the LO signals may have an 85% duty cycle and an 80% offset. In some embodiments, each branch of the mixer circuitry (e.g., the in-phase (I) and quadrature phase (Q) path) may operate at an 80% duty cycle, which may result in a significant reduction in power consumption.

The RF input signal 1407 (FIG. 14) may comprise a balanced signal, although the scope of the embodiments is not limited in this respect. The I and Q baseband output signals may be provided to low-noise amplifier, such as amplifier circuitry 1506 (FIG. 15) or to filter circuitry 1508 (FIG. 15).

In some embodiments, the output baseband signals 1507 and the input baseband signals 1511 may be analog baseband signals, although the scope of the embodiments is not limited in this respect. In some alternate embodiments, the output baseband signals 1507 and the input baseband signals 1511 may be digital baseband signals. In these alternate embodiments, the radio IC circuitry may include analog-to-digital converter (ADC) and digital-to-analog converter (DAC) circuitry.

In some dual-mode embodiments, a separate radio IC circuitry may be provided for processing signals for each spectrum, or for other spectrums not mentioned here, although the scope of the embodiments is not limited in this respect.

In some embodiments, the synthesizer circuitry 1504 may be a fractional-N synthesizer or a fractional N/N+1 synthesizer, although the scope of the embodiments is not limited in this respect as other types of frequency synthesizers may be suitable. For example, synthesizer circuitry 1504 may be a delta-sigma synthesizer, a frequency multiplier, or a synthesizer comprising a phase-locked loop with a frequency divider. According to some embodiments, the synthesizer circuitry 1504 may include digital synthesizer circuitry. An advantage of using a digital synthesizer circuitry is that, although it may still include some analog components, its footprint may be scaled down much more than the footprint of an analog synthesizer circuitry. In some embodiments, frequency input into synthesizer circuitry 1504 may be provided by a voltage controlled oscillator (VCO), although that is not a requirement. A divider control input may further be provided by either the baseband processing circuitry 1308a-b (FIG. 13) or a link aggregator depending on the desired output frequency 1505. In some embodiments, a divider control input (e.g., N) may be determined from a look-up table (e.g., within a Wi-Fi card) based on a channel number and a channel center frequency as determined or indicated by the link aggregator. The application processor 1310 is connected to the example AP-based scheduling/ACK controller 112 and/or the example STA-based scheduling/ACK controller 114 (e.g., depending on what device the radio architecture 110 is being implemented in).

In some embodiments, synthesizer circuitry 1504 may be configured to generate a carrier frequency as the output frequency 1505, while in other embodiments, the output frequency 1505 may be a fraction of the carrier frequency (e.g., one-half the carrier frequency, one-third the carrier frequency). In some embodiments, the output frequency 1505 may be a LO frequency (fLO).

FIG. 16 illustrates a functional block diagram of baseband processing circuitry 1308a in accordance with some embodiments. The baseband processing circuitry 1308a is one example of circuitry that may be suitable for use as the baseband processing circuitry 1308a (FIG. 13), although other circuitry configurations may also be suitable. Alternatively, the example of FIG. 153 may be used to implement the example BT baseband processing circuitry 1308b of FIG. 13.

The baseband processing circuitry 1308a may include a receive baseband processor (RX BBP) 1602 for processing receive baseband signals 1509 provided by the radio IC circuitry 1306a-b (FIG. 13) and a transmit baseband processor (TX BBP) 1604 for generating transmit baseband signals 1511 for the radio IC circuitry 1306a-b. The baseband processing circuitry 1308a may also include control logic 1606 for coordinating the operations of the baseband processing circuitry 1308a.

In some embodiments (e.g., when analog baseband signals are exchanged between the baseband processing circuitry 1308a-b and the radio IC circuitry 1306a-b), the baseband processing circuitry 1308a may include ADC 1610 to convert analog baseband signals 1609 received from the radio IC circuitry 1306a-b to digital baseband signals for processing by the RX BBP 1602. In these embodiments, the baseband processing circuitry 1308a may also include DAC 1612 to convert digital baseband signals from the TX BBP 1604 to analog baseband signals 1611.

In some embodiments that communicate OFDM signals or OFDMA signals, such as through baseband processor 1308a, the transmit baseband processor 1604 may be configured to generate OFDM or OFDMA signals as appropriate for transmission by performing an inverse fast Fourier transform (IFFT). The receive baseband processor 1602 may be configured to process received OFDM signals or OFDMA signals by performing an FFT. In some embodiments, the receive baseband processor 1602 may be configured to detect the presence of an OFDM signal or OFDMA signal by performing an autocorrelation, to detect a preamble, such as a short preamble, and by performing a cross-correlation, to detect a long preamble. The preambles may be part of a predetermined frame structure for Wi-Fi communication.

Referring back to FIG. 13, in some embodiments, the antennas 1301 (FIG. 13) may each comprise one or more directional or omnidirectional antennas, including, for example, dipole antennas, monopole antennas, patch antennas, loop antennas, microstrip antennas or other types of antennas suitable for transmission of RF signals. In some multiple-input multiple-output (MIMO) embodiments, the antennas may be effectively separated to take advantage of spatial diversity and the different channel characteristics that may result. Antennas 1301 may each include a set of phased-array antennas, although embodiments are not so limited.

Although the radio-architecture 110 is illustrated as having several separate functional elements, one or more of the functional elements may be combined and may be implemented by

combinations of software-configured elements, such as processing elements including digital signal processors (DSPs), and/or other hardware elements. For example, some elements may comprise one or more microprocessors, DSPs, field-programmable gate arrays (FPGAs), application specific integrated circuits (ASICs), radio-frequency integrated circuits (RFICs) and combinations of various hardware and logic circuitry for performing at least the functions described herein. In some embodiments, the functional elements may refer to one or more processes operating on one or more processing elements.

FIG. 17 is a block diagram of an example processor platform 1700 capable of executing the instructions of FIGS. 5-11 to implement the example AP-based scheduling/ACK controller 112 and/or the example STA-based scheduling/ACK controller 114 of FIGS. 1 and 2. The processor platform 1700 can be, for example, a server, a personal computer, a mobile device (e.g., a cell phone, a smart phone, a tablet such as an iPadTM), a personal digital assistant (PDA), an Internet appliance, or any other type of computing device.

The processor platform 1700 of the illustrated example includes a processor 1712. The processor 1712 of the illustrated example is hardware. For example, the processor 1712 can be implemented by integrated circuits, logic circuits, microprocessors or controllers from any desired family or manufacturer.

The processor 1712 of the illustrated example includes a local memory 1713 (e.g., a cache). The example processor 1712 of FIG. 17 executes the instructions of FIGS. 5-12 to implement the example component interface 200, the example interval tracker 202, the example semi-static scheduler 204, the example packet generator 206, and the example ACK protocol processor 208 of the example AP-based scheduling/ACK controller 112 or the example component interface 300, the example interval tracker 302, the example packet processor 304, the example packet generator 306, the example semi-static schedule database 308, and the example ACK protocol processor 310 of the example STA-based scheduling/ACK controller 114.

The processor 1712 of the illustrated example is in communication with a main memory including a volatile memory 1714 and a non-volatile memory 1716 via a bus 1718. The volatile memory 1714 may be implemented by Synchronous Dynamic Random Access Memory (SDRAM), Dynamic Random Access Memory (DRAM), RAMBUS Dynamic Random Access Memory (RDRAM) and/or any other type of random access memory device. The non-volatile

memory 1716 may be implemented by flash memory and/or any other desired type of memory device. Access to the main memory 1714, 1716 is controlled by a clock controller.

The processor platform 1700 of the illustrated example also includes an interface circuit 1720. The interface circuit 1720 may be implemented by any type of interface standard, such as
5 an Ethernet interface, a universal serial bus (USB), and/or a PCI express interface.

In the illustrated example, one or more input devices 1722 are connected to the interface circuit 1720. The input device(s) 1722 permit(s) a user to enter data and commands into the processor 1712. The input device(s) can be implemented by, for example, a sensor, a microphone, a camera (still or video), a keyboard, a button, a mouse, a touchscreen, a track-pad,
10 a trackball, isopoint and/or a voice recognition system.

One or more output devices 1724 are also connected to the interface circuit 1720 of the illustrated example. The output devices 1724 can be implemented, for example, by display devices (e.g., a light emitting diode (LED), an organic light emitting diode (OLED), a liquid crystal display, a cathode ray tube display (CRT), a touchscreen, a tactile output device, and/or
15 speakers). The interface circuit 1720 of the illustrated example, thus, typically includes a graphics driver card, a graphics driver chip or a graphics driver processor.

The interface circuit 1720 of the illustrated example also includes a communication device such as a transmitter, a receiver, a transceiver, a modem and/or network interface card to facilitate exchange of data with external machines (e.g., computing devices of any kind) via a
20 network 1726 (e.g., an Ethernet connection, a digital subscriber line (DSL), a telephone line, coaxial cable, a cellular telephone system, etc.).

The processor platform 1700 of the illustrated example also includes one or more mass storage devices 1728 for storing software and/or data. Examples of such mass storage devices 1728 include floppy disk drives, hard drive disks, compact disk drives, Blu-ray disk drives,
25 RAID systems, and digital versatile disk (DVD) drives.

The coded instructions 1732 of FIGS. 5-11 may be stored in the mass storage device 1728, in the volatile memory 1714, in the non-volatile memory 1716, and/or on a removable tangible computer readable storage medium such as a CD or DVD.

Example 1 includes an apparatus to facilitate semi-static scheduling, the apparatus
30 comprising a semi-static scheduler to determine if two or more transmission intervals correspond to a same transmission characteristic in a wireless local area network, and a packet generator to

during a first transmission interval of the two or more transmission intervals, generate a first data packet including (a) a first value identifying when the two or more transmission intervals will occur and (b) a second value identifying the transmission characteristic.

Example 2 includes the apparatus of example 1, wherein the packet generator is to,
5 during a subsequent transmission interval of the two or more transmission intervals, generate a second data packet omitting the first value and the second value.

Example 3 includes the apparatus of example 2, further including an interval tracker to track transmission intervals to determine when the subsequent transmission interval occurs.

Example 4 includes the apparatus of example 2, wherein, when the first transmission
10 interval corresponds to uplink transmissions, the first and second data packets are trigger frames.

Example 5 includes the apparatus of example 2, wherein, when the first transmission interval corresponds correspond to downlink transmissions, the first and second data packets are downlink data packets.

Example 6 includes the apparatus of example 1, wherein the two or more transmission
15 intervals are within a transmission opportunity.

Example 7 includes the apparatus of example 1, wherein the two or more transmission intervals are across two or more transmission opportunities.

Example 8 includes the apparatus of example 1, wherein the transmission characteristic is at least one of a type of transmission or a resource allocation, the type of transmission being
20 uplink or downlink.

Example 9 includes the apparatus of example 1, wherein the packet generator is to generate the first data packet to include a third value corresponding to whether the two or more transmission intervals are within a transmission opportunity or across two or more transmission opportunities.

Example 10 includes a tangible computer readable storage medium comprising
25 instructions which, when executed, cause a machine to at least determine if two or more transmission intervals correspond to a same transmission characteristic in a wireless local area network, and during a first transmission interval of the two or more transmission intervals, generate a first data packet including (a) a first value identifying when the two or more
30 transmission intervals will occur and (b) a second value identifying the transmission characteristic.

Example 11 includes the computer readable storage medium of example 10, wherein the instructions cause the machine to, during a subsequent transmission interval of the two or more transmission intervals, generate a second data packet omitting the first value and the second value.

5 Example 12 includes the computer readable storage medium of example 11, wherein the instructions cause the machine to track transmission intervals to determine when the subsequent transmission interval occurs.

10 Example 13 includes the computer readable storage medium of example 11, wherein, when the first transmission interval corresponds to uplink transmissions, the first and second data packets are trigger frames.

Example 14 includes the computer readable storage medium of example 11, wherein, when the first transmission interval corresponds correspond to downlink transmissions, the first and second data packets are downlink data packets.

15 Example 15 includes the computer readable storage medium of example 10, wherein the two or more transmission intervals are within a transmission opportunity.

Example 16 includes the computer readable storage medium of example 10, wherein the two or more transmission intervals are across two or more transmission opportunities.

20 Example 17 includes the computer readable storage medium of example 10, wherein the transmission characteristic is at least one of a type of transmission or a resource allocation, the type of transmission being uplink or downlink.

Example 18 includes the computer readable storage medium of example 10, wherein the instructions cause the machine to generate the first data packet to include a third value corresponding to whether the two or more transmission intervals are within a transmission opportunity or across two or more transmission opportunities.

25 Example 19 includes a method to facilitate semi-static scheduling, the method comprising determining if two or more transmission intervals correspond to a same transmission characteristic in a wireless local area network, and during a first transmission interval of the two or more transmission intervals, generating a first data packet including (a) a first value identifying when the two or more transmission intervals will occur and (b) a second value
30 identifying the transmission characteristic.

Example 20 includes the method of example 19, further including, during a subsequent transmission interval of the two or more transmission intervals, generating a second data packet omitting the first value and the second value.

5 Example 21 includes the method of example 20, further including tracking transmission intervals to determine when the subsequent transmission interval occurs.

Example 22 includes the method of example 20, wherein, when the first transmission interval corresponds to uplink transmissions, the first and second data packets are trigger frames.

10 Example 23 includes the method of example 20, when the first transmission interval corresponds correspond to downlink transmissions, the first and second data packets are downlink data packets.

Example 24 includes the method of example 19, wherein the two or more transmission intervals are within a transmission opportunity.

Example 25 includes the method of example 19, wherein the two or more transmission intervals are across two or more transmission opportunities.

15 Example 26 includes the method of example 19, wherein the transmission characteristic is at least one of a type of transmission or a resource allocation, the type of transmission being uplink or downlink.

20 Example 27 includes the method of example 19, further including generating the first data packet to include a third value corresponding to whether the two or more transmission intervals are within a transmission opportunity or across two or more transmission opportunities.

25 Example 28 includes an apparatus to facilitate an acknowledgement protocol, the apparatus comprising a packet generator to generate a first data frame corresponding to downlink data, the first data frame including information linking a first resource unit of a frequency band with a station, the first data frame to be included in a preamble for a transmission opportunity, and generate a second data frame corresponding to uplink data, the second data frame including information linking a second resource unit of the frequency band with the station, the second data frame to be included in a first delayed-acknowledgement corresponding to the uplink data, a component interface to, when the uplink data is received across two or more transmission intervals, transmit the first delayed-acknowledgement corresponding to first uplink data
30 transmitted by the station using the second resource unit, and an acknowledgement protocol processor to, when the downlink data is received across two or more transmission intervals, infer

that a received second delayed-acknowledgement corresponds to first downlink data from the station based on being received at the first resource unit.

Example 29 includes the apparatus of example 10, wherein the component interface is to transmit the first delayed-acknowledgement by instruction radio architecture to transmit the first
5 delayed-acknowledgement.

Example 30 includes the apparatus of example 10, wherein the first resource unit is the second resource unit.

Example 31 includes the apparatus of example 10, wherein the first data frame includes an acknowledgement type.

10 Example 32 includes the apparatus of example 10, wherein the station is a first station, the first data frame including a second linking of a third resource unit of the frequency band with a second station, the second data frame includes a third linking of a fourth resource unit of the frequency band with the second station, and the second data frame to be included in a third delayed-acknowledgment.

15 Example 33 includes the apparatus of example 14, wherein the component interface is to, when the uplink data is received across the two or more transmission intervals, transmit the third delayed-acknowledgment corresponding to second uplink data transmitted by the second station using the fourth resource unit, and the acknowledgment protocol processor is to, when the downlink data is received across the two or more transmission intervals, infer that a received
20 fourth delayed-acknowledgement corresponds to second downlink data from the second station based on being received at the third resource unit.

Example 34 includes the apparatus of example 10, wherein the first delayed-acknowledgement is a physical layer protocol data unit.

25 Although certain example methods, apparatus and articles of manufacture have been described herein, the scope of coverage of this patent is not limited thereto. On the contrary, this patent covers all methods, apparatus and articles of manufacture fairly falling within the scope of the claims of this patent.

What Is Claimed Is:

1. An apparatus to facilitate semi-static scheduling, the apparatus comprising:
a semi-static scheduler to determine if two or more transmission intervals correspond to a same transmission characteristic in a wireless local area network; and
5 a packet generator to during a first transmission interval of the two or more transmission intervals, generate a first data packet including (A) a first value identifying when the two or more transmission intervals will occur and (B) a second value identifying the transmission characteristic.
2. The apparatus of claim 1, wherein the packet generator is to, during a subsequent
10 transmission interval of the two or more transmission intervals, generate a second data packet omitting the first value and the second value.
3. The apparatus of claim 2, further including an interval tracker to track transmission intervals to determine when the subsequent transmission interval occurs.
4. The apparatus of claim 2, wherein, when the first transmission interval
15 corresponds to uplink transmissions, the first and second data packets are trigger frames.
5. The apparatus of claim 2, wherein, when the first transmission interval corresponds correspond to downlink transmissions, the first and second data packets are downlink data packets.
6. The apparatus of claim 1, wherein the two or more transmission intervals are
20 within a transmission opportunity.
7. The apparatus of claim 1, wherein the two or more transmission intervals are across two or more transmission opportunities.
8. The apparatus of claim 1, wherein the transmission characteristic is at least one of
25 a type of transmission or a resource allocation, the type of transmission being uplink or downlink.
9. The apparatus of claim 1, wherein the packet generator is to generate the first data packet to include a third value corresponding to whether the two or more transmission intervals are within a transmission opportunity or across two or more transmission opportunities.
10. A tangible computer readable storage medium comprising instructions which,
30 when executed, cause a machine to at least:

determine if two or more transmission intervals correspond to a same transmission characteristic in a wireless local area network; and

during a first transmission interval of the two or more transmission intervals, generate a first data packet including (A) a first value identifying when the two or more transmission intervals will occur and (B) a second value identifying the transmission characteristic.

11. The computer readable storage medium of claim 10, wherein the instructions cause the machine to, during a subsequent transmission interval of the two or more transmission intervals, generate a second data packet omitting the first value and the second value.

12. The computer readable storage medium of claim 11, wherein the instructions cause the machine to track transmission intervals to determine when the subsequent transmission interval occurs.

13. The computer readable storage medium of claim 11, wherein, when the first transmission interval corresponds to uplink transmissions, the first and second data packets are trigger frames.

14. The computer readable storage medium of claim 11, wherein, when the first transmission interval corresponds correspond to downlink transmissions, the first and second data packets are downlink data packets.

15. The computer readable storage medium of claim 10, wherein the two or more transmission intervals are within a transmission opportunity.

16. The computer readable storage medium of claim 10, wherein the two or more transmission intervals are across two or more transmission opportunities.

17. The computer readable storage medium of claim 10, wherein the transmission characteristic is at least one of a type of transmission or a resource allocation, the type of transmission being uplink or downlink.

18. The computer readable storage medium of claim 10, wherein the instructions cause the machine to generate the first data packet to include a third value corresponding to whether the two or more transmission intervals are within a transmission opportunity or across two or more transmission opportunities.

19. A method to facilitate semi-static scheduling, the method comprising:
determining if two or more transmission intervals correspond to a same transmission characteristic in a wireless local area network; and

during a first transmission interval of the two or more transmission intervals, generating a first data packet including (A) a first value identifying when the two or more transmission intervals will occur and (B) a second value identifying the transmission characteristic.

20. The method of claim 19, further including, during a subsequent transmission
5 interval of the two or more transmission intervals, generating a second data packet omitting the first value and the second value.

21. The method of claim 20, further including tracking transmission intervals to determine when the subsequent transmission interval occurs.

22. The method of claim 20, wherein, when the first transmission interval
10 corresponds to uplink transmissions, the first and second data packets are trigger frames.

23. The method of claim 20, when the first transmission interval corresponds correspond to downlink transmissions, the first and second data packets are downlink data packets.

24. The method of claim 19, wherein the two or more transmission intervals are
15 within a transmission opportunity.

25. The method of claim 19, wherein the two or more transmission intervals are across two or more transmission opportunities.

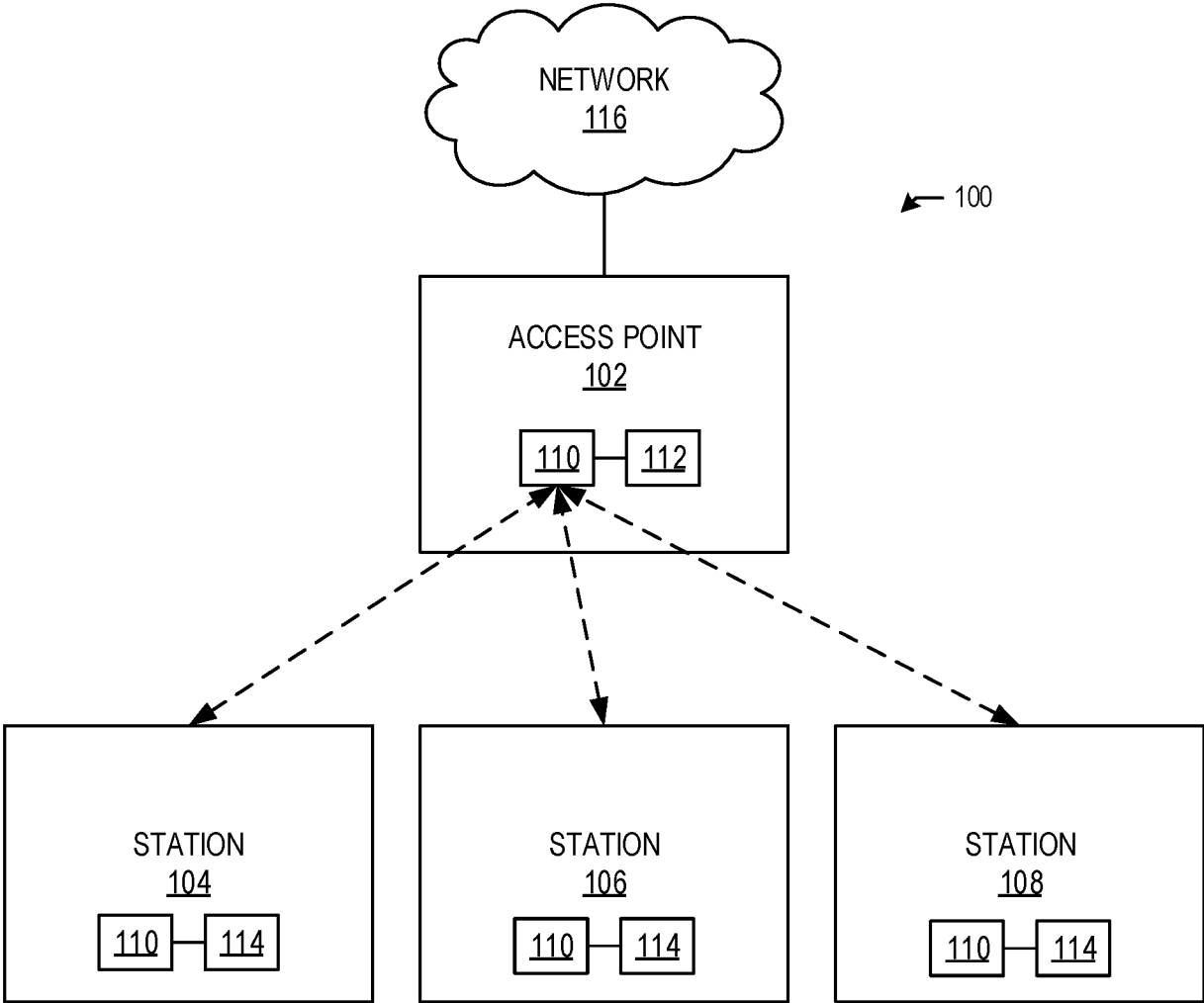


FIG. 1

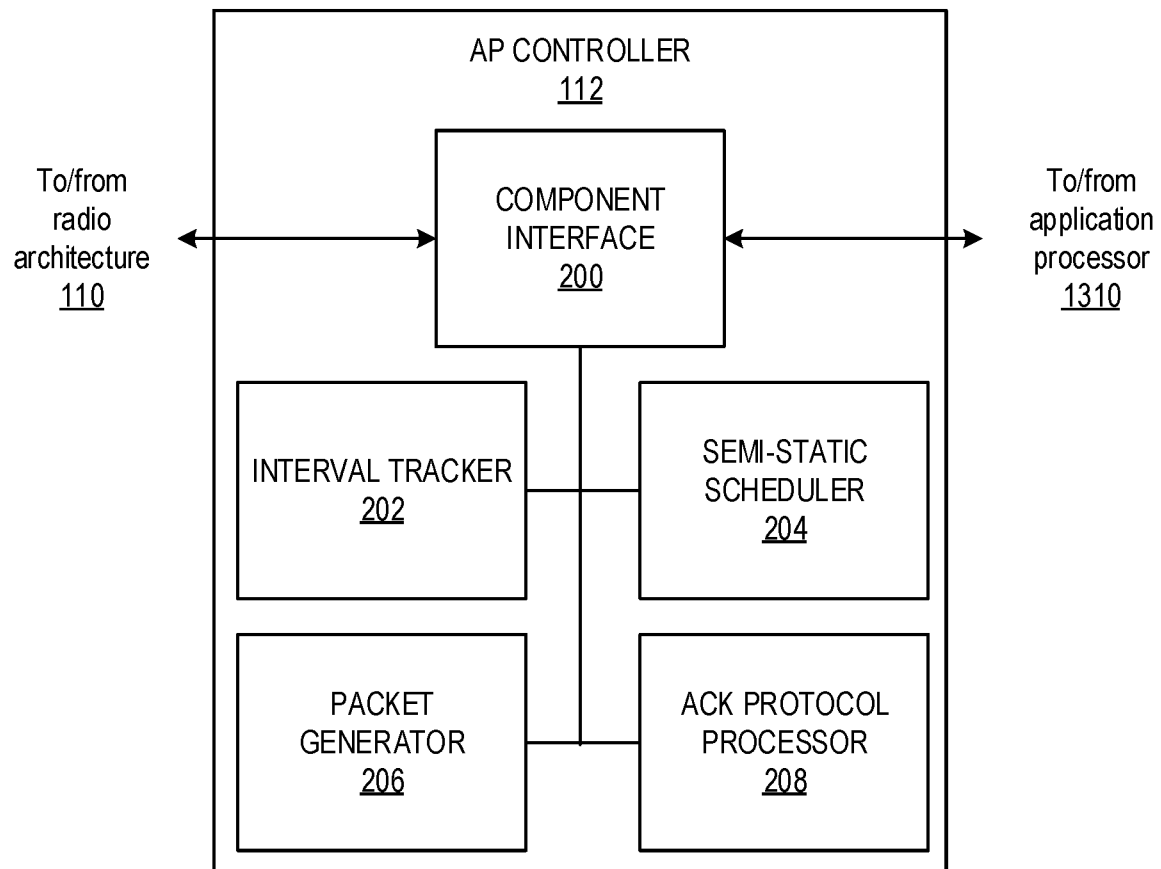


FIG. 2

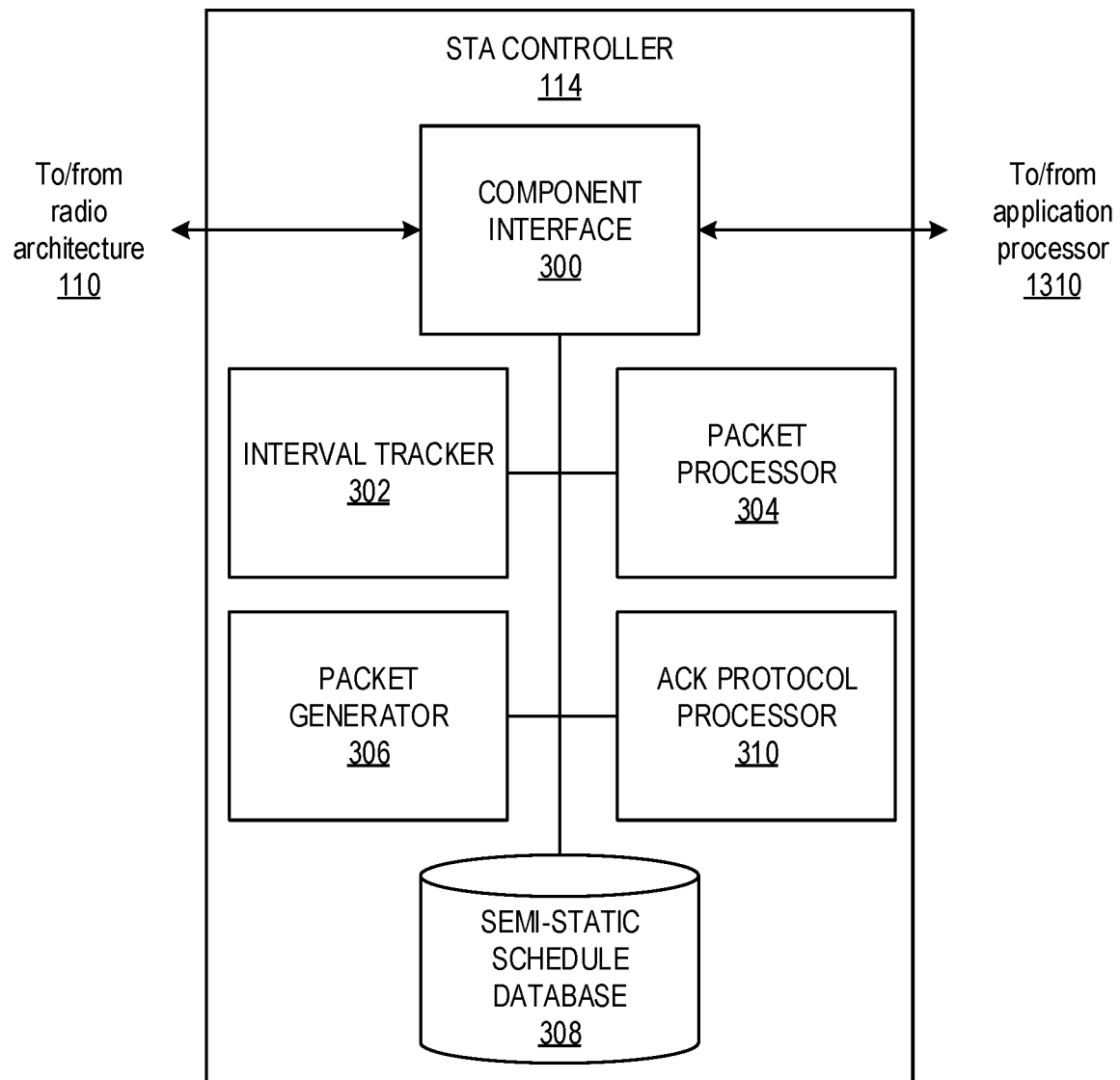


FIG. 3

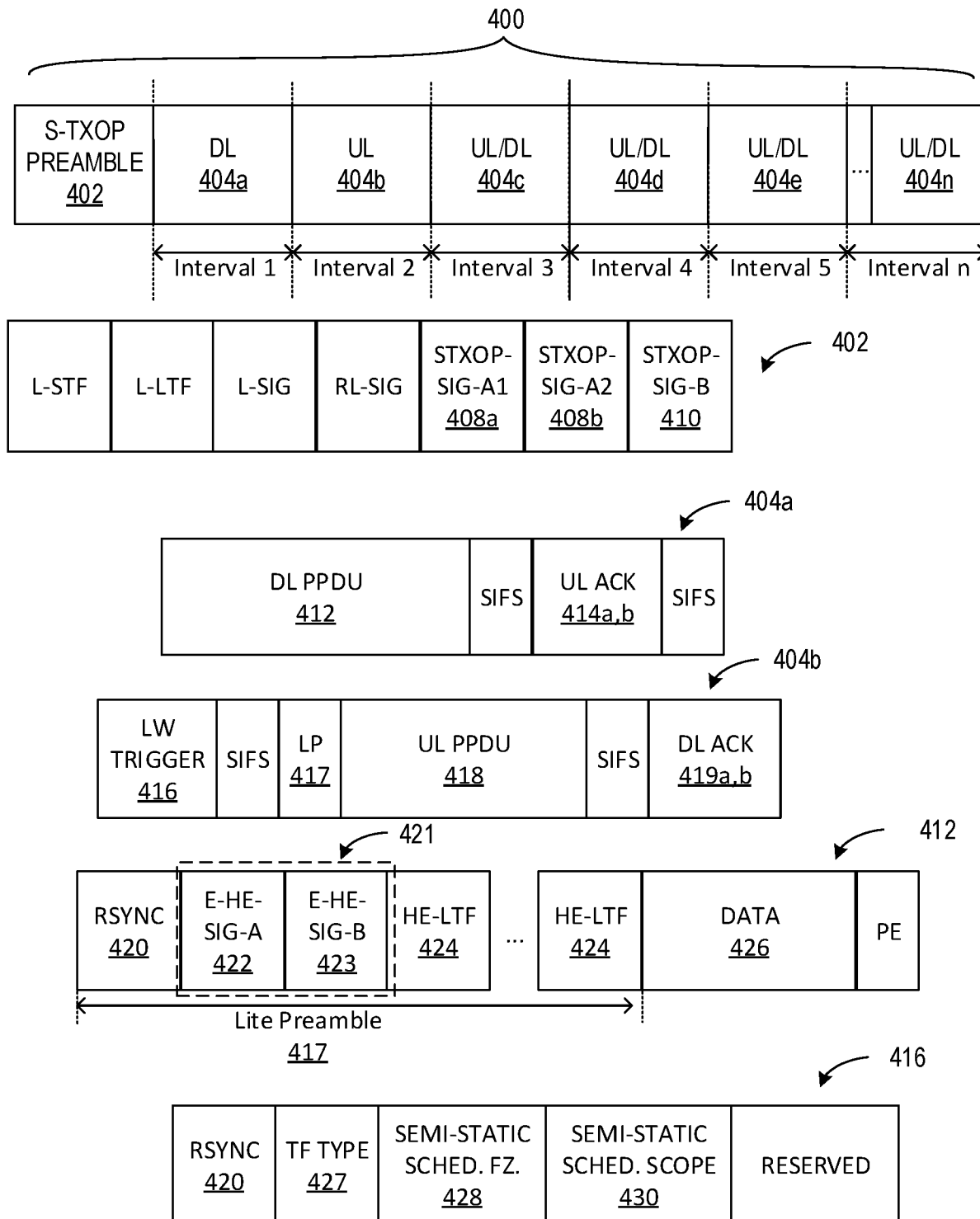


FIG. 4A

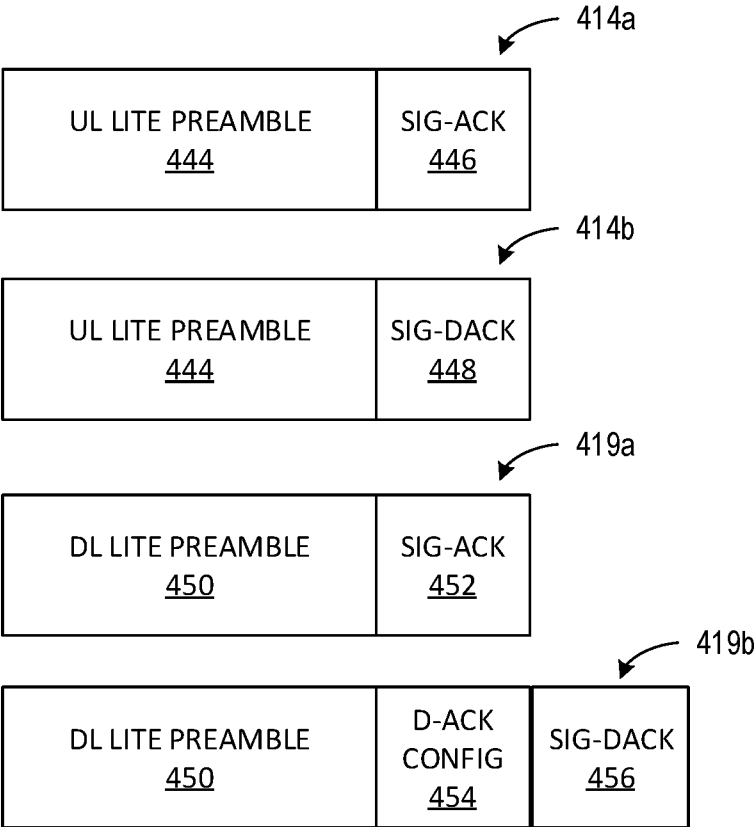


FIG. 4B

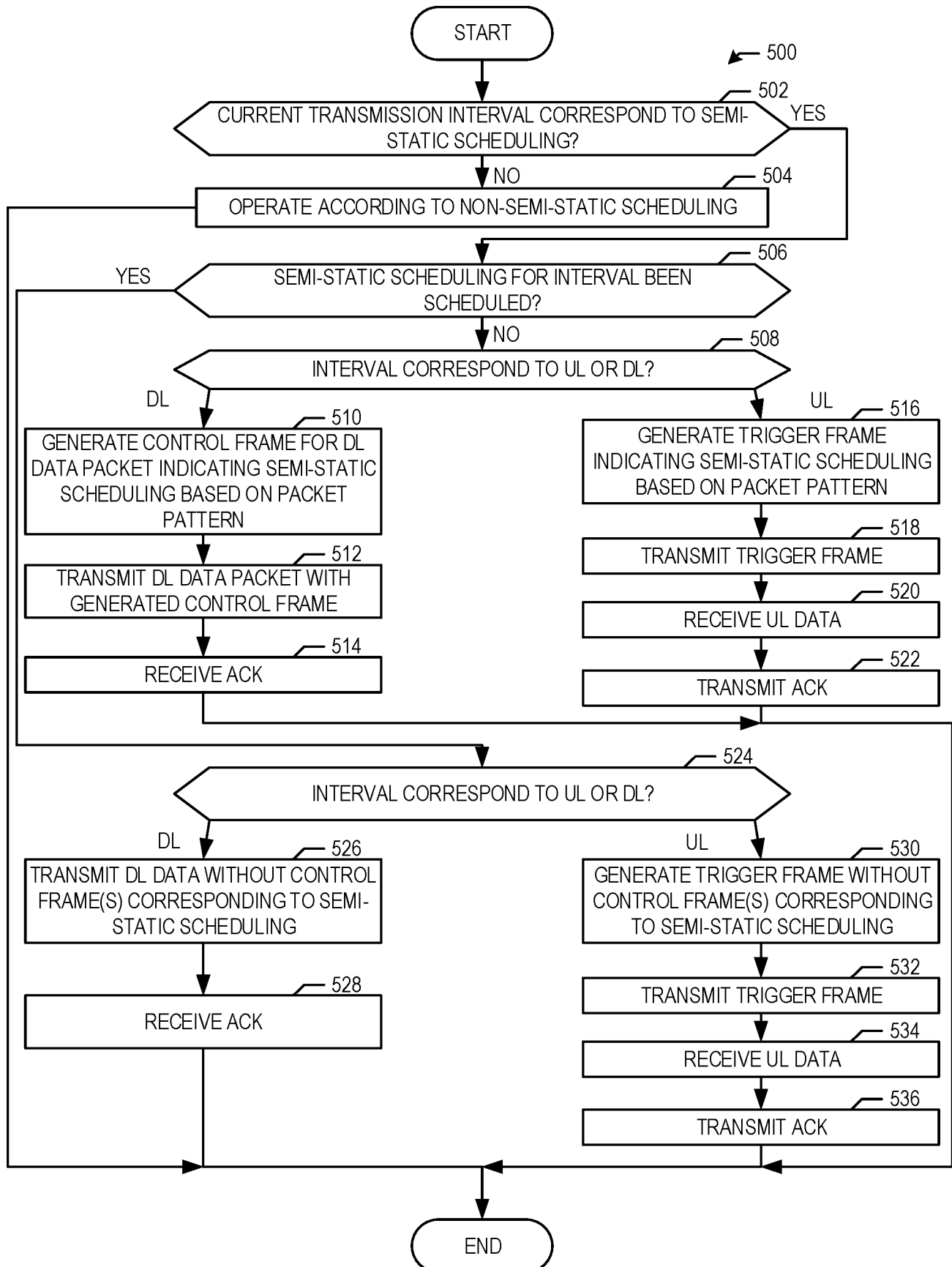


FIG. 5

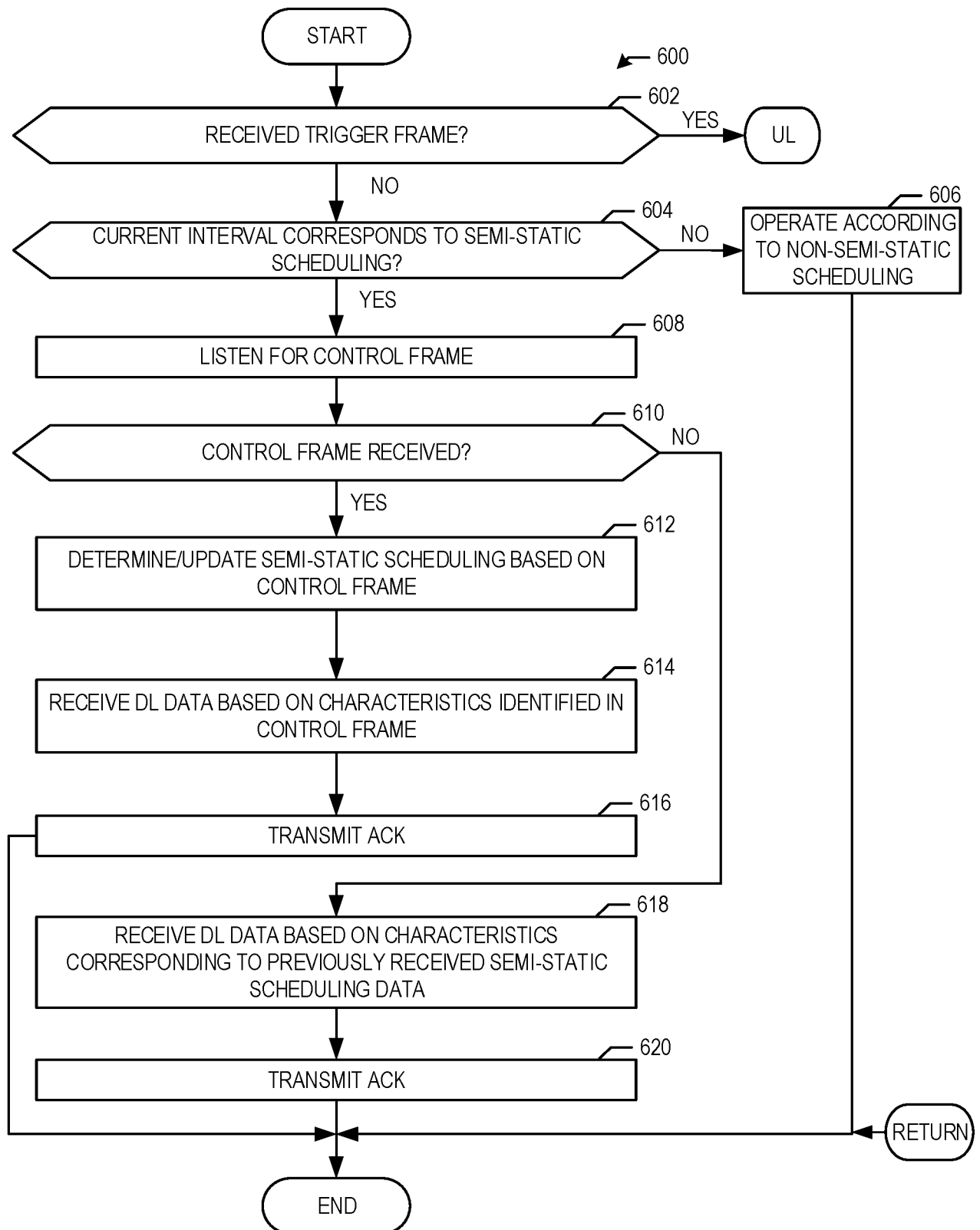


FIG. 6

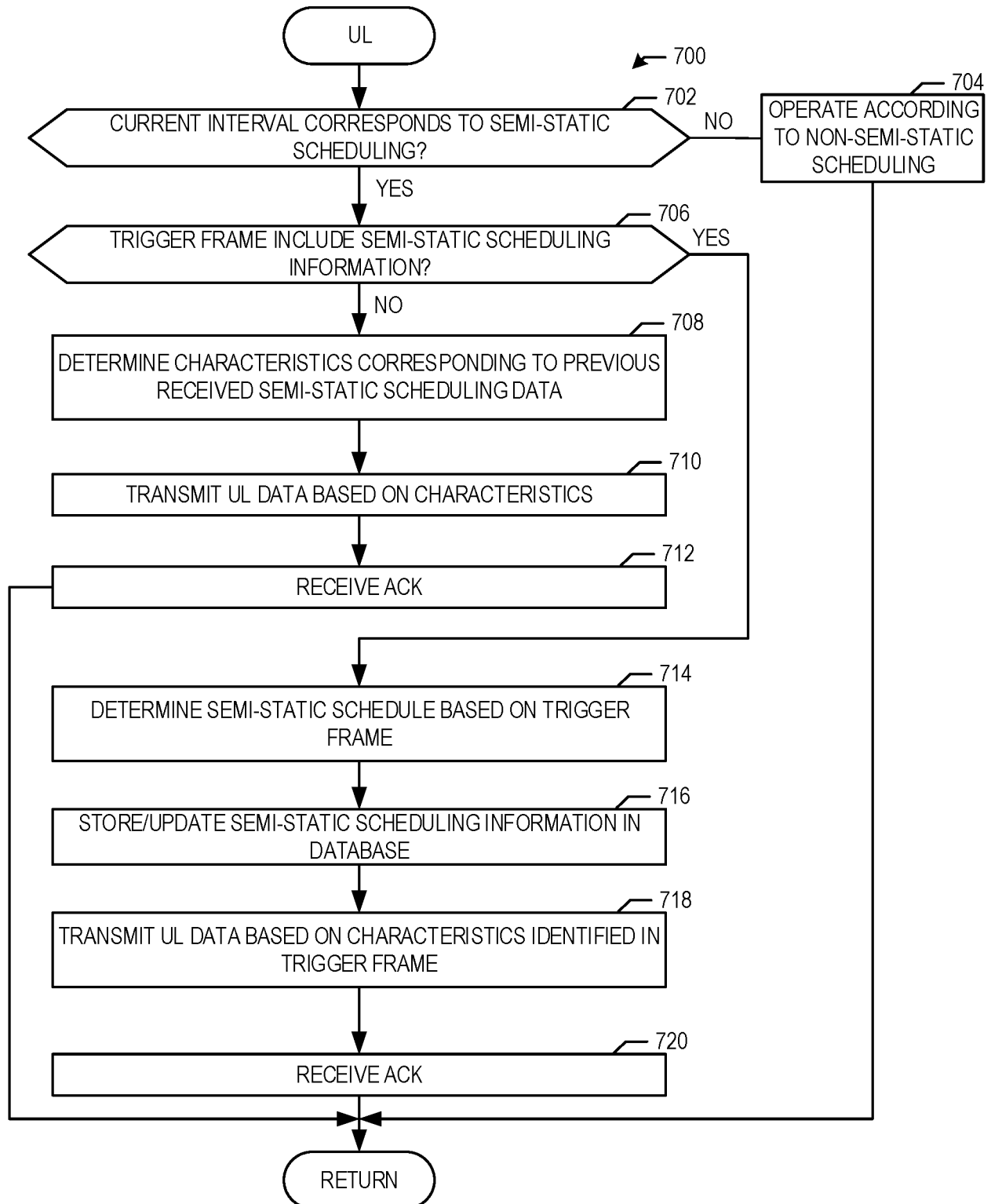


FIG. 7

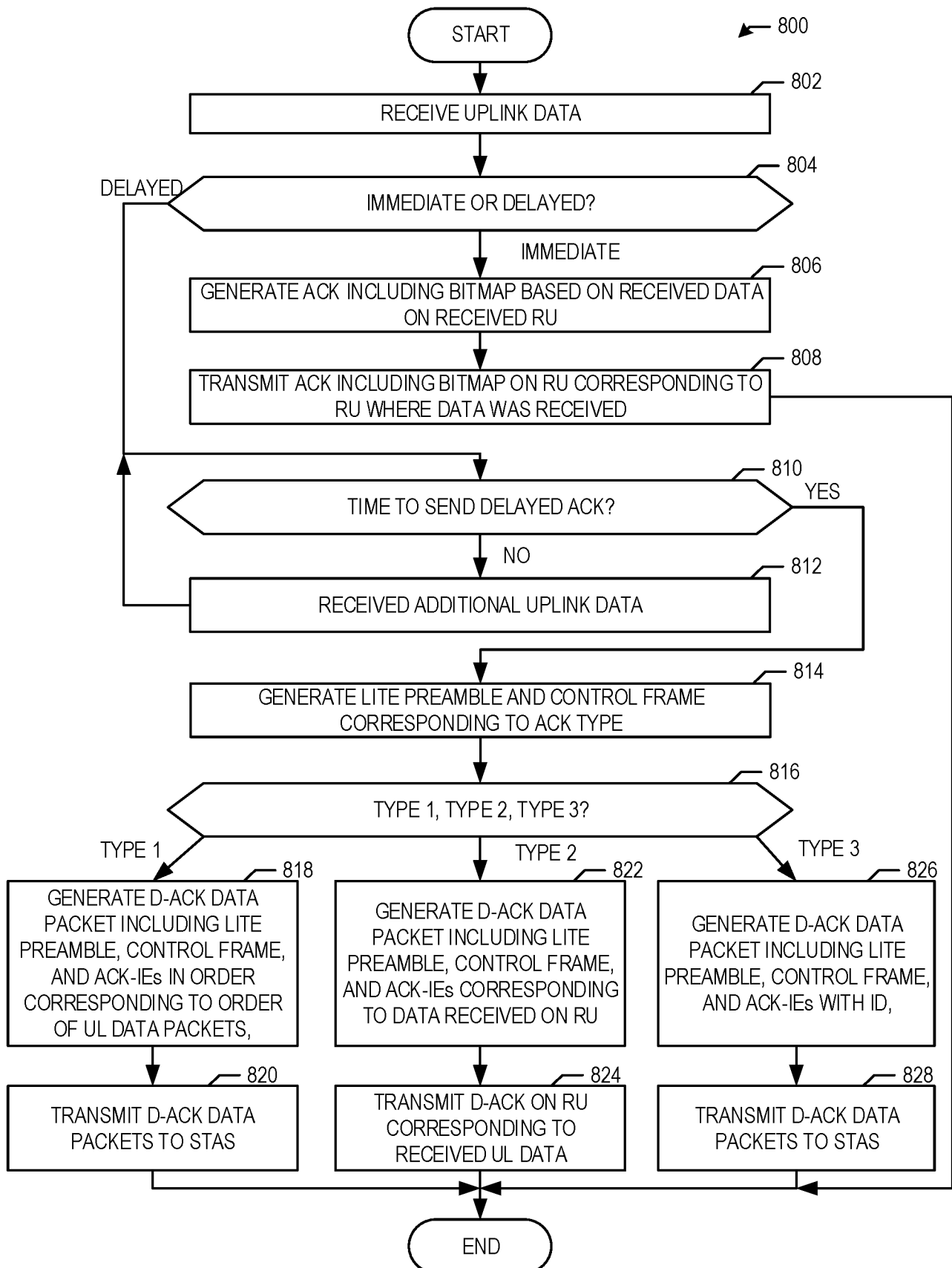


FIG. 8

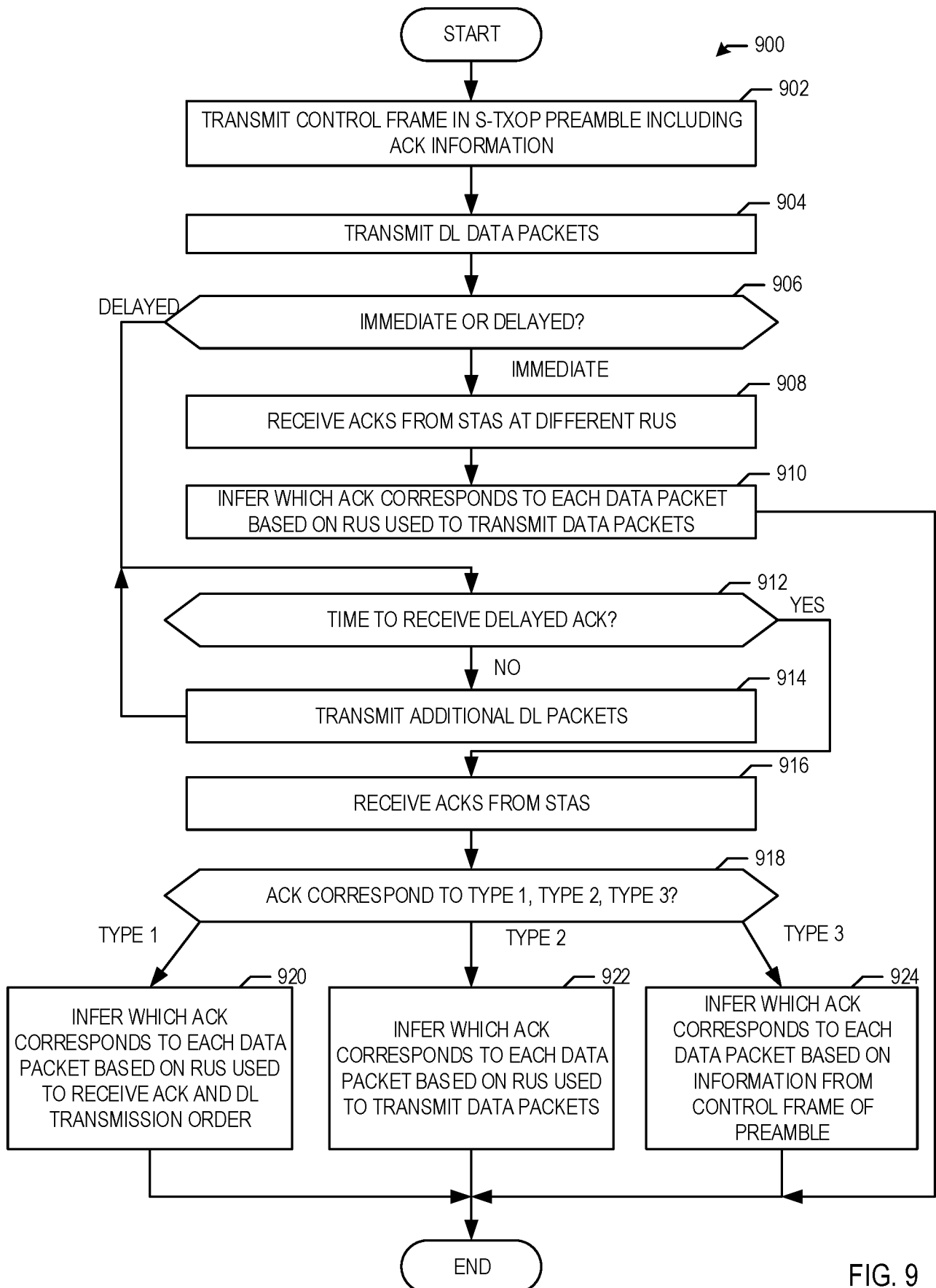


FIG. 9

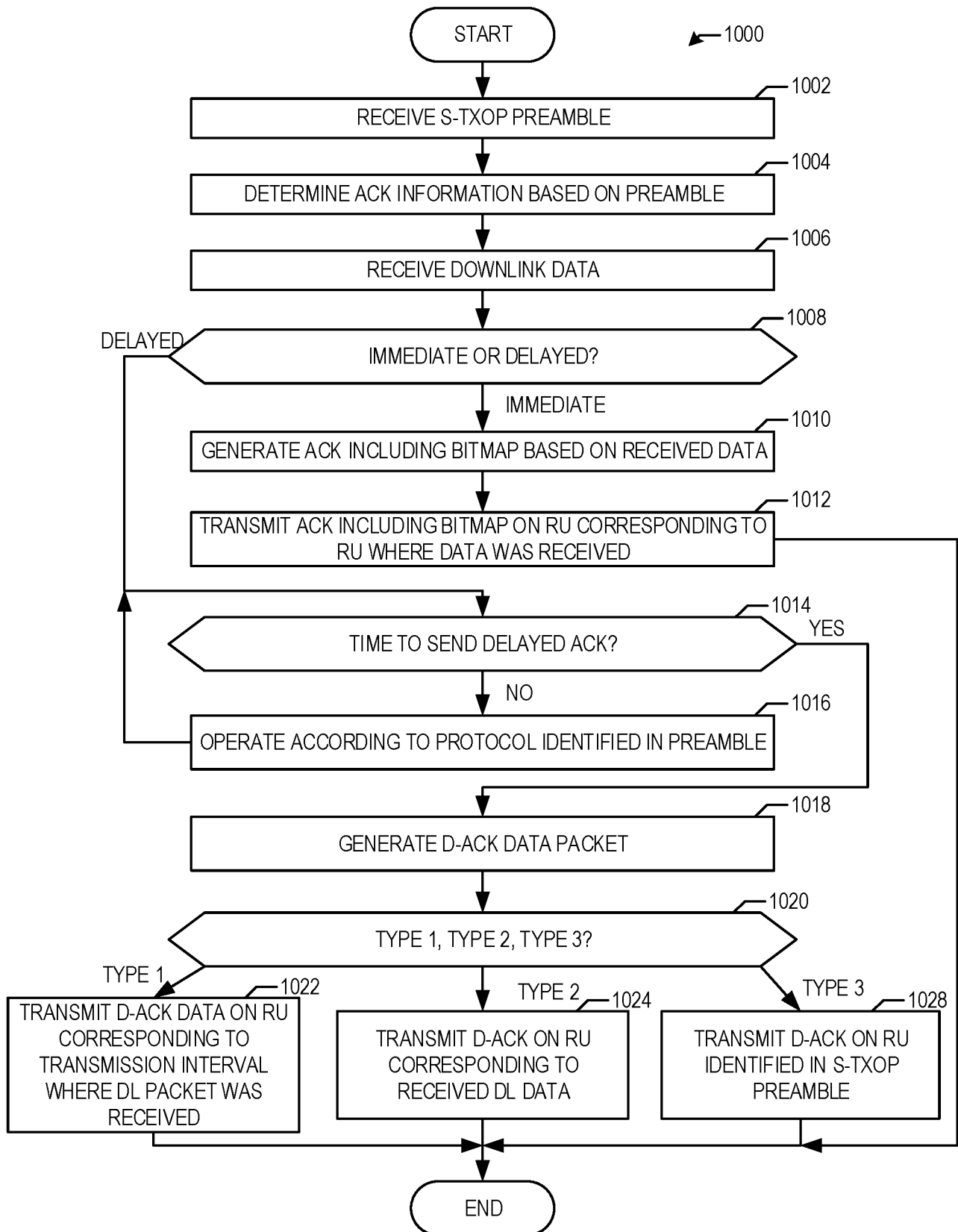


FIG. 10

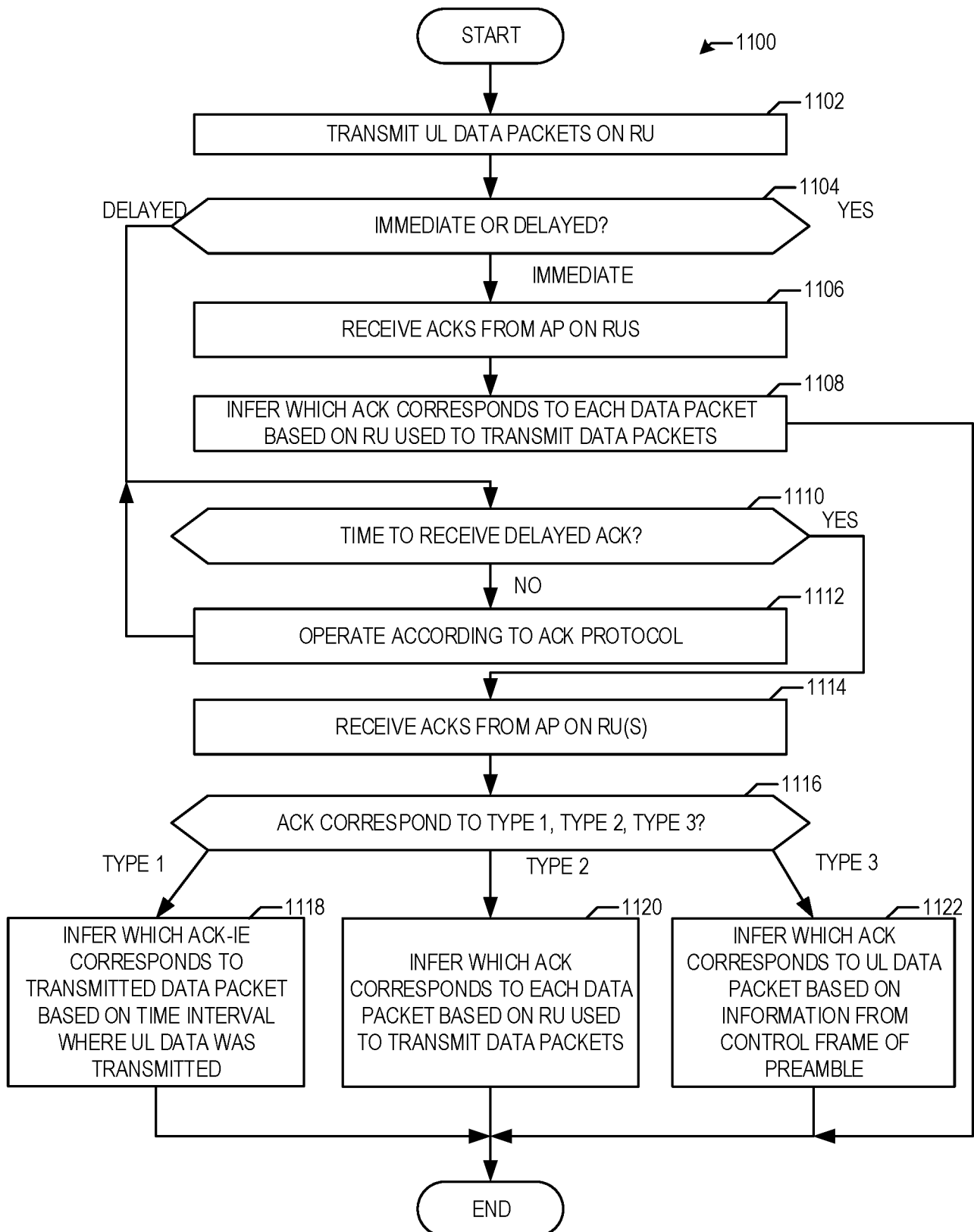


FIG. 11

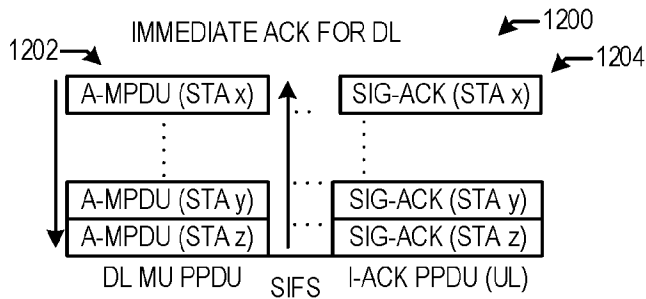


FIG. 12A

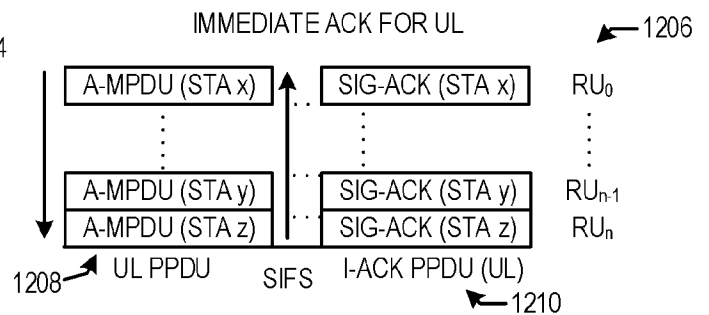


FIG. 12B

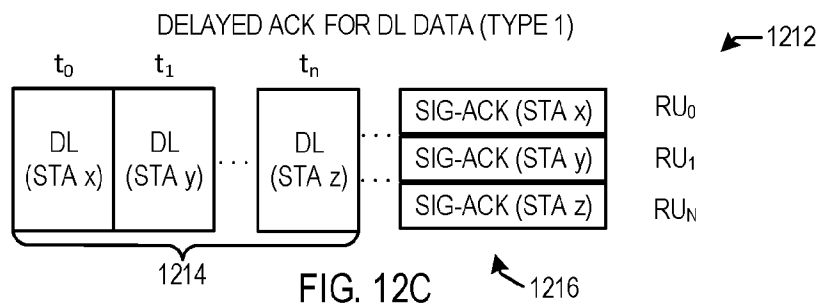


FIG. 12C

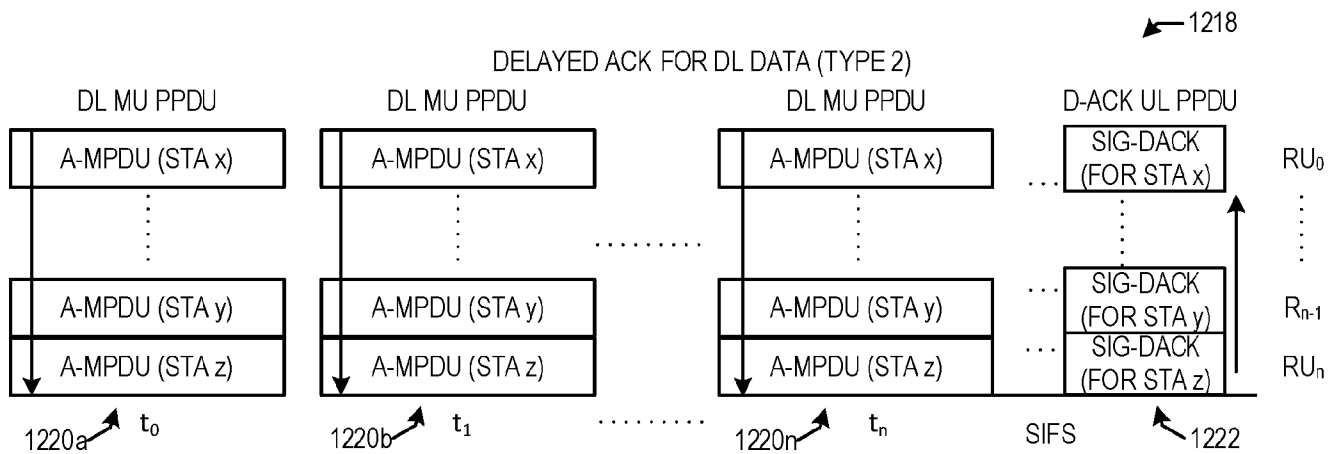


FIG. 12D

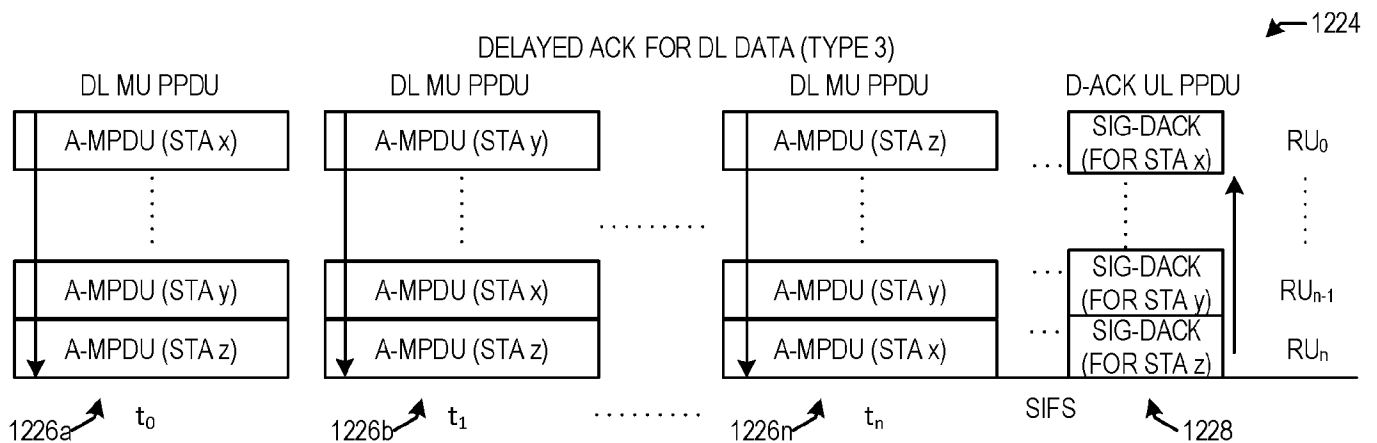


FIG. 12E

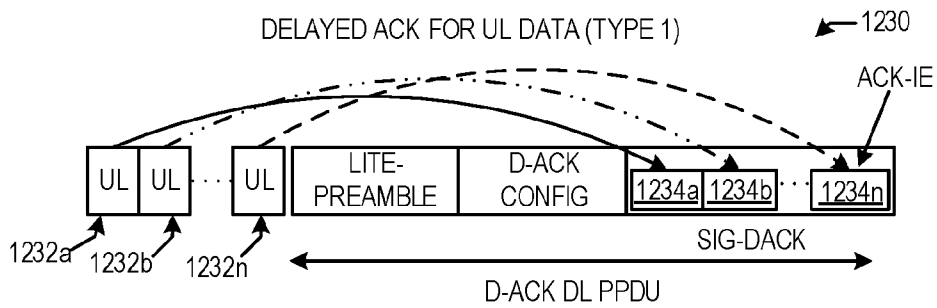


FIG. 12F

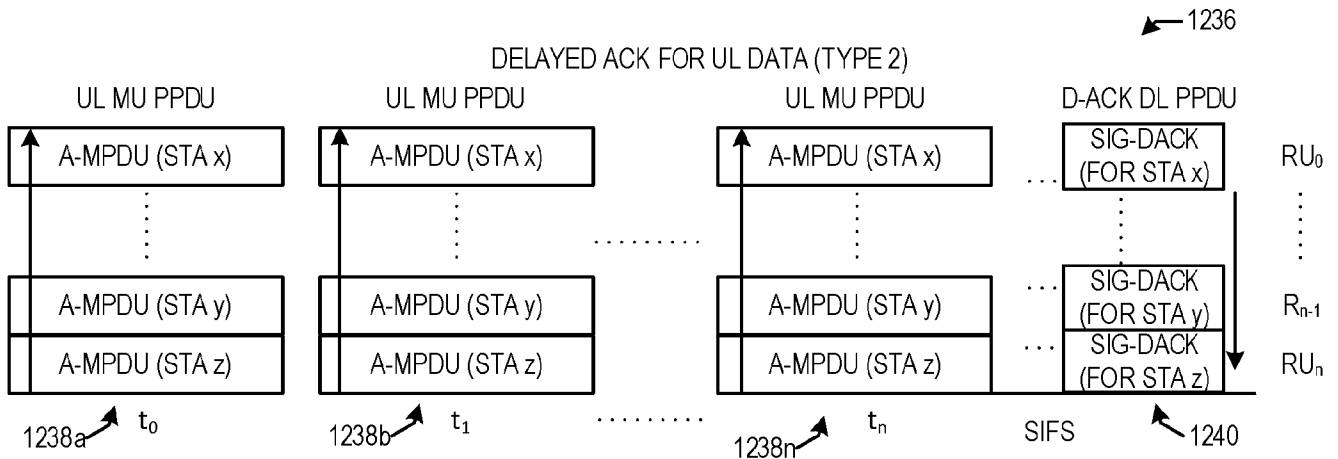


FIG. 12G

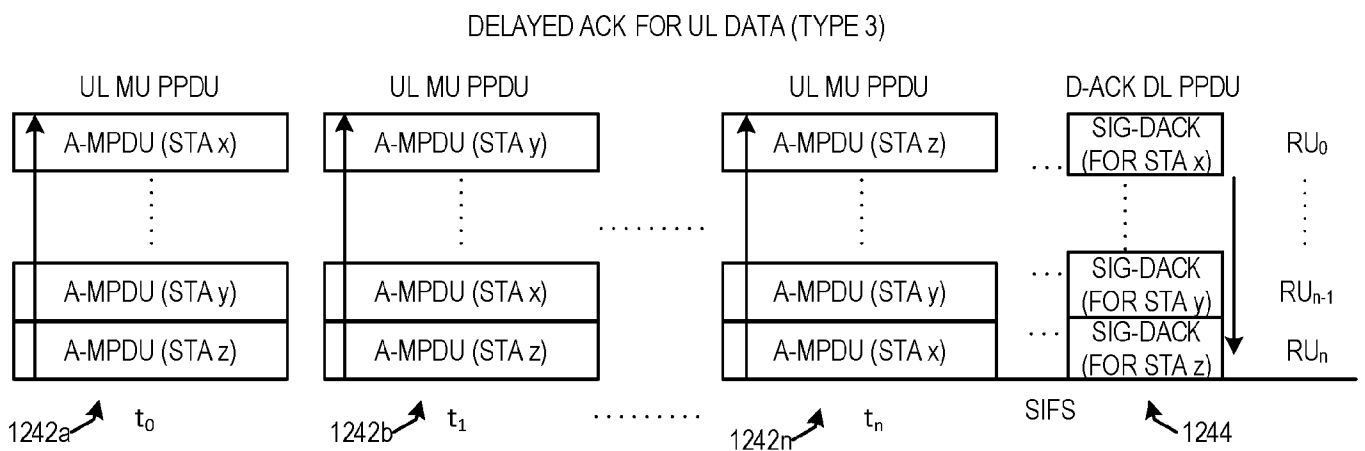


FIG. 12H

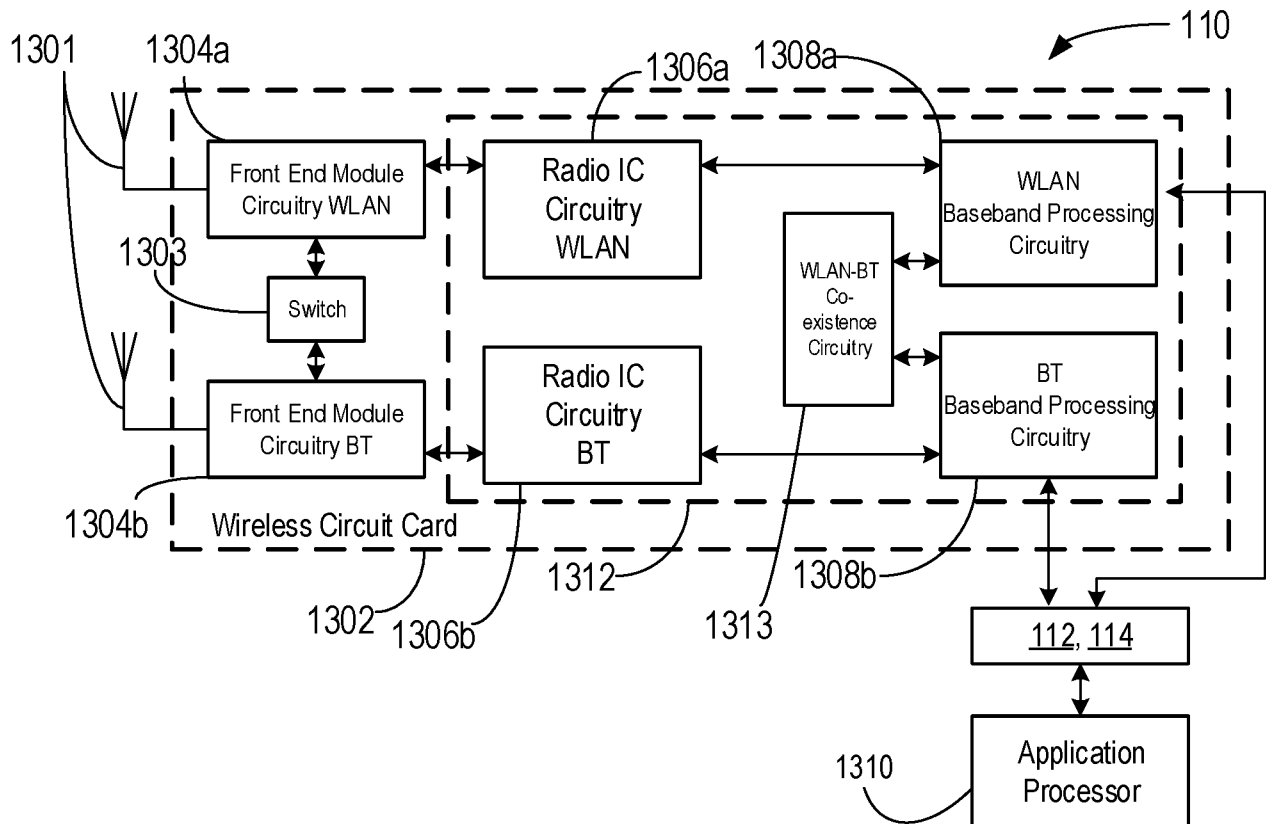


FIG. 13

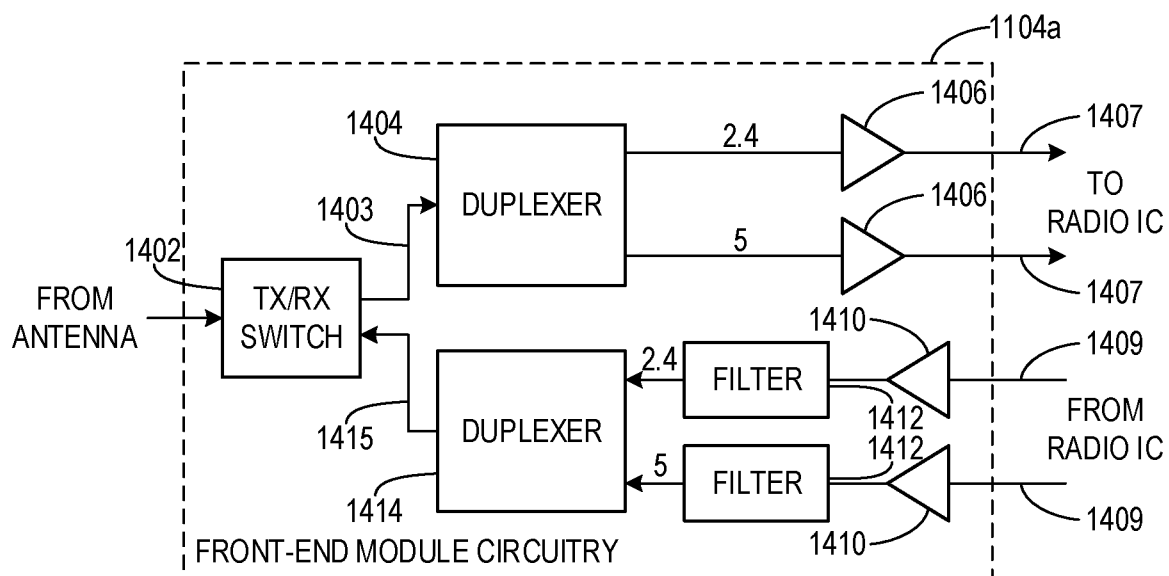


FIG. 14

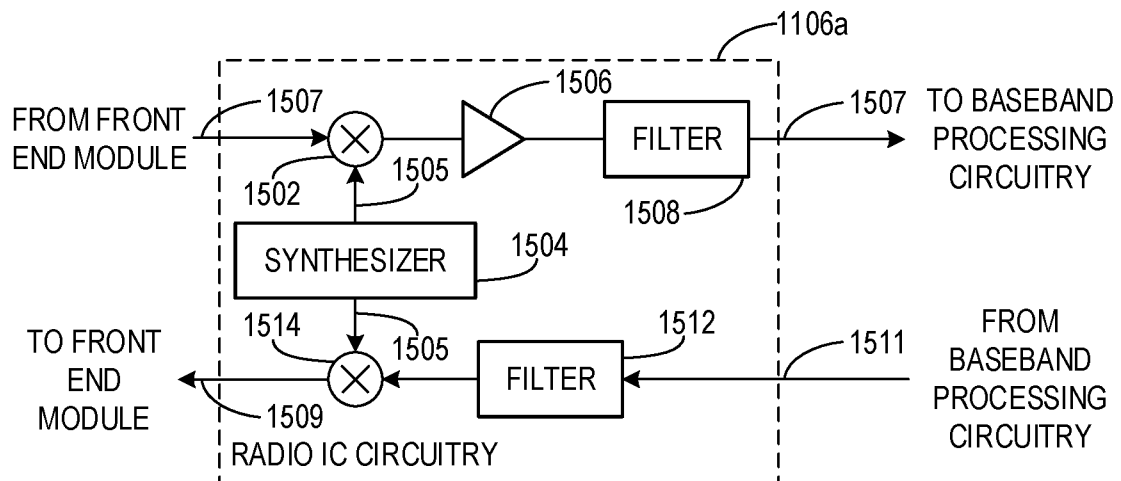


FIG. 15

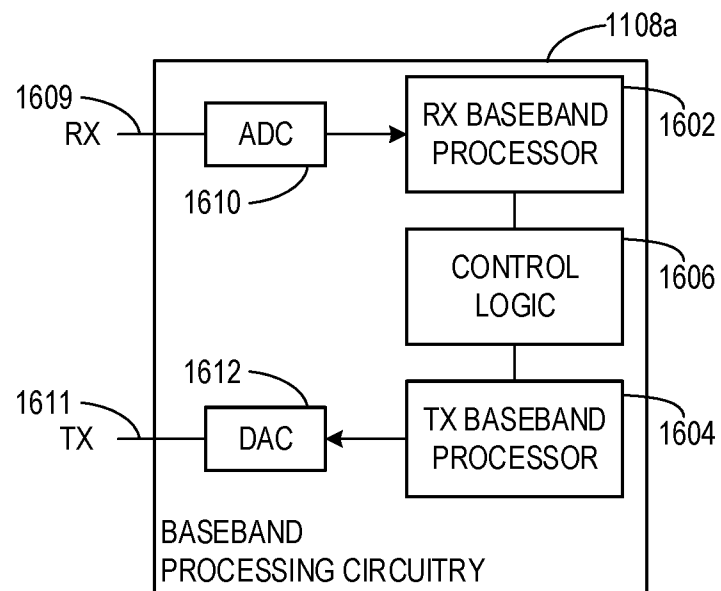


FIG. 16

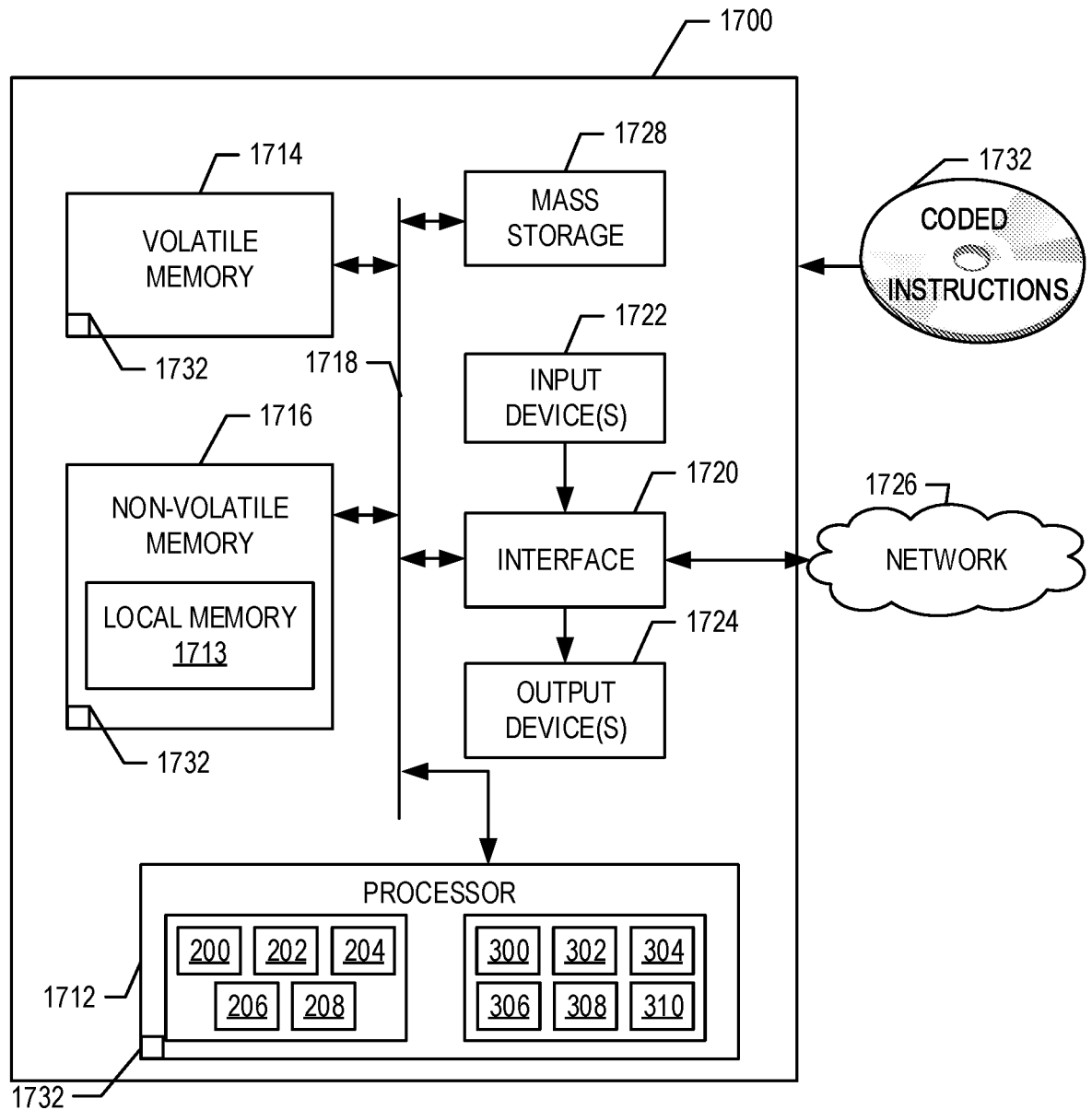


FIG. 17

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2018/039308**A. CLASSIFICATION OF SUBJECT MATTER****H04W 72/12(2009.01)i, H04W 84/12(2009.01)i**

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

H04W 72/12; H04B 7/00; H04B 7/216; H04J 3/00; H04J 3/24; H04L 12/18; H04L 12/26; H04Q 7/00; H04W 84/12

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Korean utility models and applications for utility models

Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

eKOMPASS(KIPO internal) & keywords: semi-static scheduler, transmission, interval, uplink, downlink, packet

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X A	US 2008-0080465 A1 (KARI PAJUKOSKI et al.) 03 April 2008 See paragraphs [0036]-[0054]; claim 11; and figures 5A-6.	1,6-8,10,15-17,19 ,24,25 2-5,9,11-14,18 ,20-23
A	US 2009-0213750 A1 (ALEXEI GOROKHOV et al.) 27 August 2009 See paragraphs [0048]-[0067]; and figures 5-8.	1-25
A	US 2013-0156034 A1 (TOMOYUKI ONO et al.) 20 June 2013 See paragraphs [0072]-[0078]; and figure 8.	1-25
A	US 2005-0078651 A1 (JU-HO LEE et al.) 14 April 2005 See paragraphs [0096]-[0099]; and figure 13.	1-25
A	US 6775254 B1 (SERGE WILLENEGGER et al.) 10 August 2004 See column 13, line 21 - column 16, line 19; and figures 6A-7B.	1-25



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

21 March 2019 (21.03.2019)

Date of mailing of the international search report

21 March 2019 (21.03.2019)

Name and mailing address of the ISA/KR

International Application Division

Korean Intellectual Property Office

189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea



Facsimile No. +82-42-481-8578

Authorized officer

KANG, Hee Gok



Telephone No. +82-42-481-8264

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2018/039308

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2008-0080465 A1	03/04/2008	CN 101523971 A EP 2067330 A1 FI 20065614 A WO 2008-037855 A1	02/09/2009 10/06/2009 30/03/2008 03/04/2008
US 2009-0213750 A1	27/08/2009	AU 2006-282879 A1 AU 2006-282879 B2 BR PI0615061 A2 CA 2620217 A1 CA 2620217 C CL 2006002230 A1 CL 22302006 A1 CN 101292456 A CN 101297575 A CN 101297575 B CN 102065561 A CN 102082650 A CN 102523082 A CN 102523082 B CN 102523083 A CN 102523083 B CN 105337698 A DE 06813760 T1 EP 1920624 A2 EP 1938493 A2 EP 1938493 B1 EP 2665219 A1 ES 2306637 T1 ES 2306637 T3 HK 1169232 A1 HK 1169527 A1 IL 189703 A IL 189703 B IL 189703 D0 JP 2009-506679 A JP 2009-506726 A JP 2011-229151 A JP 2012-085315 A JP 4891323 B2 JP 5059958 B2 JP 5301634 B2 KR 10-1029946 B1 KR 10-1045732 B1 KR 10-1061745 B1 KR 10-2008-0032656 A KR 10-2008-0049786 A KR 10-2010-0050560 A MY 151145 A NO 20081320 A	01/03/2007 30/09/2010 03/05/2011 01/03/2007 28/05/2013 24/03/2008 24/03/2008 22/10/2008 29/10/2008 01/05/2013 18/05/2011 01/06/2011 27/06/2012 25/11/2015 27/06/2012 16/09/2015 17/02/2016 24/12/2008 14/05/2008 02/07/2008 08/10/2014 20/11/2013 16/11/2008 23/01/2015 17/06/2016 24/03/2016 31/12/2012 31/12/2012 03/08/2009 12/02/2009 12/02/2009 10/11/2011 26/04/2012 07/03/2012 31/10/2012 25/09/2013 19/04/2011 30/06/2011 02/09/2011 15/04/2008 04/06/2008 13/05/2010 30/04/2014 22/05/2008

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2018/039308

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		NZ 566202 A	30/07/2010
		RU 2008-110963 A	27/09/2009
		RU 2010-110520 A	27/09/2011
		RU 2407179 C2	20/12/2010
		RU 2462823 C2	27/09/2012
		TW 200718253 A	01/05/2007
		TW 200723813 A	16/06/2007
		TW 201029420 A	01/08/2010
		TW I340604 B	11/04/2011
		TW I370657 B	11/08/2012
		TW I427997 B	21/02/2014
		UA 94059 C2	11/04/2011
		US 2007-0047485 A1	01/03/2007
		US 2007-0047495 A1	01/03/2007
		US 8644292 B2	04/02/2014
		US 8787347 B2	22/07/2014
		WO 2007-025160 A2	01/03/2007
		WO 2007-025160 A3	18/05/2007
		WO 2007-027733 A2	08/03/2007
		WO 2007-027733 A3	26/04/2007
US 2013-0156034 A1	20/06/2013	CN 102387066 A	21/03/2012
		CN 102387066 B	04/08/2017
		CN 202750106 U	20/02/2013
		EP 2613481 A1	10/07/2013
		EP 2613481 A4	02/11/2016
		JP 2013-029575 A1	28/10/2013
		JP 5776692 B2	09/09/2015
		US 9166804 B2	20/10/2015
		WO 2012-029575 A1	08/03/2012
US 2005-0078651 A1	14/04/2005	CN 1604687 A	06/04/2005
		EP 1509011 A2	23/02/2005
		EP 1509011 A3	19/01/2011
		JP 2005-065298 A	10/03/2005
		JP 4243229 B2	25/03/2009
		KR 10-0703131 B1	05/04/2007
		KR 10-2005-0018540 A	23/02/2005
US 6775254 B1	10/08/2004	AT 515840 T	15/07/2011
		AU 2002-009002 A	21/05/2002
		AU 2010-249181 A1	23/12/2010
		AU 220090 B2	23/08/2007
		BR 0115216 A	09/05/2006
		BR PI0115216 B1	10/05/2016
		CA 2428965 A1	16/05/2002
		CA 2428965 C	05/02/2013
		CN 100486546 A	31/03/2004
		CN 100576774 C	30/12/2009
		CN 101714894 A	26/05/2010

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2018/039308Patent document
cited in search reportPublication
datePatent family
member(s)Publication
date

CN 101714894 B	02/04/2014
EP 1332569 A2	06/08/2003
EP 1332569 B1	06/07/2011
EP 2207280 A2	14/07/2010
EP 2207280 A3	27/10/2010
EP 2290848 A1	02/03/2011
EP 2290848 B1	30/08/2017
ES 2367101 T3	28/10/2011
HK 1061759 A1	17/09/2010
IL 155732 A	11/02/2009
IL 155732 D0	23/11/2003
IL 193877 D0	04/05/2009
JP 2004-536468 A	02/12/2004
JP 2007-282203 A	25/10/2007
JP 2010-233227 A	14/10/2010
JP 2013-066182 A	11/04/2013
JP 2013-118641 A	13/06/2013
JP 4068455 B2	26/03/2008
JP 5450229 B2	26/03/2014
JP 5475094 B2	16/04/2014
KR 10-0899961 B1	28/05/2009
KR 10-0899962 B1	28/05/2009
KR 10-0930845 B1	10/12/2009
KR 10-2003-0044077 A	02/06/2003
KR 10-2007-0087235 A	27/08/2007
KR 10-2009-0008477 A	21/01/2009
MX PA03004094 A	12/02/2004
NO 20032063 A	25/06/2003
NO 20032063 D0	08/05/2003
RU 2006134821 A	10/04/2008
RU 2293441 C2	10/02/2007
RU 2434339 C2	20/11/2011
TW 588529 B	21/05/2004
US 2004-0240401 A1	02/12/2004
US 8363576 B2	29/01/2013
WO 02-039595 A3	17/10/2002
WO 02-39595 A2	16/05/2002