Packet format of BLE

Write by Alen Xiao in Telink Semiconductor, 2017-03-06

1. Architecture of BT/BLE

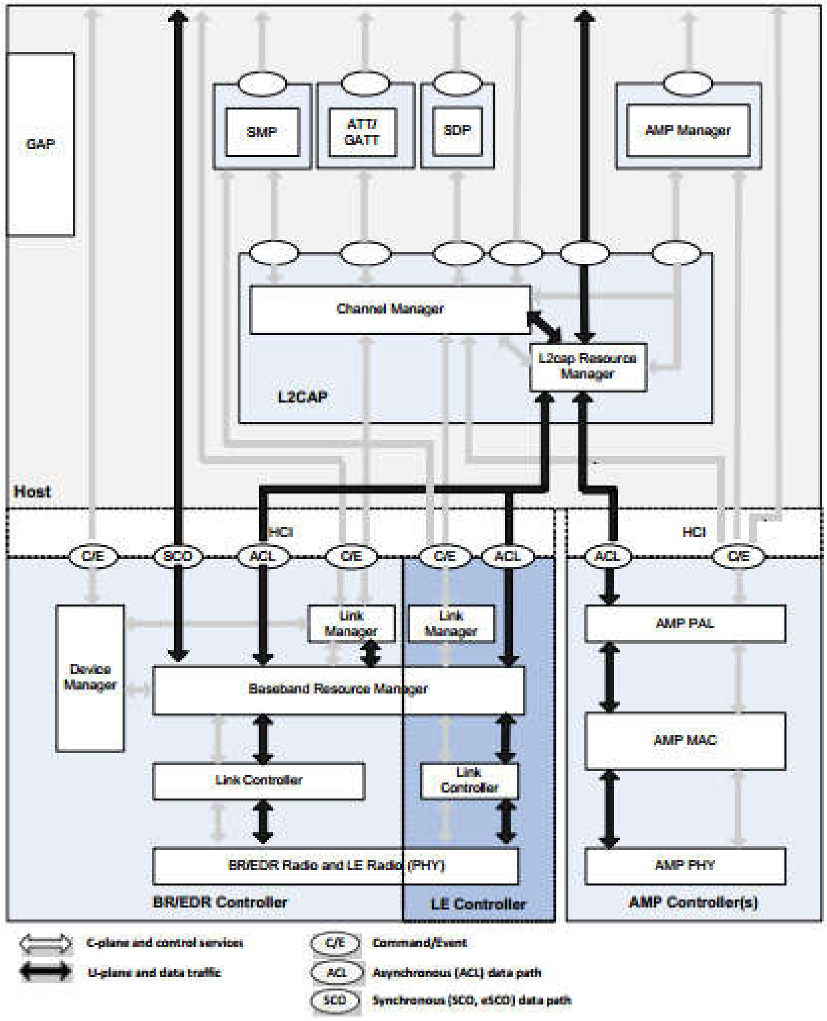


Figure 2.1: Bluetooth core system architecture

GAP: General Access Protocol

GATT: General Attribute Protocol

BLE: Bluetooth Low Energy

PAL: Protocol Adapter Layer

HCI: Host Control Interface

ATT: Attribute

LL: Link Layer

MIC: Message Integrity Check (Encrypt)

SMP: Secure Management Protocol

AMP: Alternate MAC/Phy Protocol

BR/EDR: Basic Rate/Enhanced Data Rate

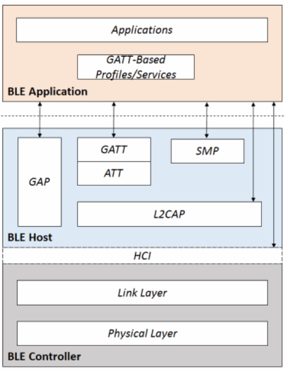
ACL: Asynchronizing Connection-oriented Logical Link

SCO: Synchronizing Connection-Oriented

L2CAP: Logical Link Control and Adapter Protocol

SDP: Service Discover Protocol

2. BLE Architecture:



OR

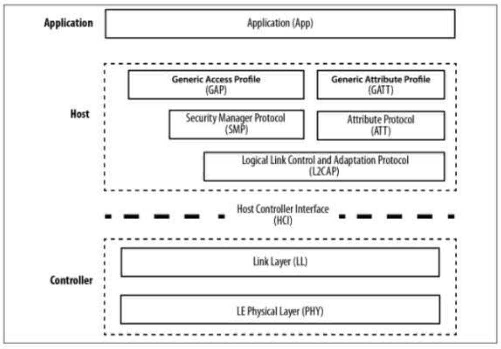

Application
Application (App)
Host
Generic Access Profile (GAP)
Generic Attribute Profile (GATT)
Security Manager Protocol (SMP)
Attribute Protocol (ATT)
Logical Link Control and Adaptation Protocol (L2CAP)
Host Controller Interface (HCI)
Controller
Link Layer (LL)
LE Physical Layer (PHY)

Figure 2-1. The BLE protocol stack

3. Packet Format & Data Tranport Architecture

3.1 Data Transport Architecture

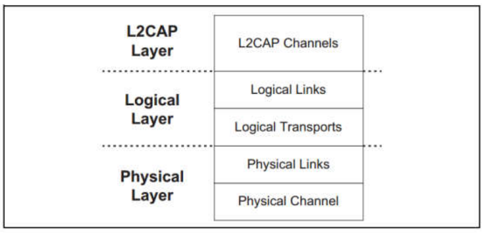
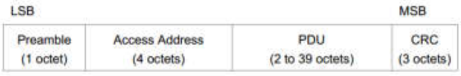


Figure 3.1: Bluetooth generic data transport architecture

3.2 Mapping between Packet Format and Channel



LSB			MSB
Preamble (1 octet)	Access Address (4 octets)	PDU (2 to 39 octets)	CRC (3 octets)

Figure 2.1: Link Layer packet format

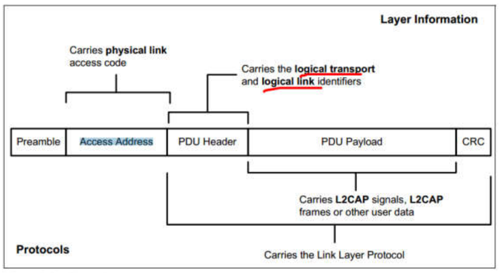

Layer Information
Carries physical link access code
Carries the logical transport and logical link identifiers
Preamble
Access Address
PDU Header
PDU Payload
CRC
Carries L2CAP signals, L2CAP frames or other user data
Protocols
Carries the Link Layer Protocol

Figure 3.5: LE packet structure

- Access Address: represents the physical link
  - ➢ The physical channel identifier is not contained in the link layer air interface packet, and physical channel identifiers are either fixed or are determined at connection setup.
  - ➢ Access Address is used to identify communications on a physical link, and to exclude or ignore packets on different physical links that are using the same PHY channels in

physical proximity (one physical channel supports multiple physical links)

- PDU Header:
  - ➢ Advertising Packet Header

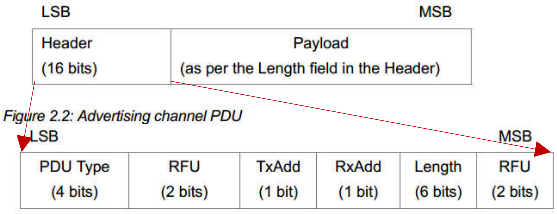
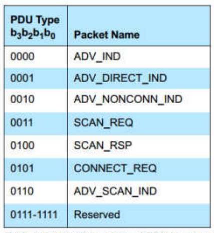


Figure 2.2: Advertising channel PDU

Figure 2.3: Advertising channel PDU Header

  - ➢ PDU Type: (Logical Link Identify) represents the logical Link



PDU Type $b_3b_2b_1b_0$	Packet Name
0000	ADV_IND
0001	ADV_DIRECT_IND
0010	ADV_NONCONN_IND
0011	SCAN_REQ
0100	SCAN_RSP
0101	CONNECT_REQ
0110	ADV_SCAN_IND
0111-1111	Reserved

Table 2.1: Advertising channel PDU Header's PDU Type field encoding

- ➢ Data Packet Header

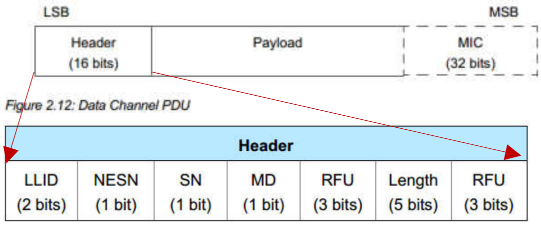
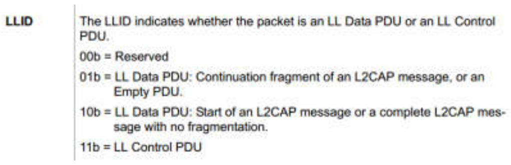


Figure 2.12: Data Channel PDU

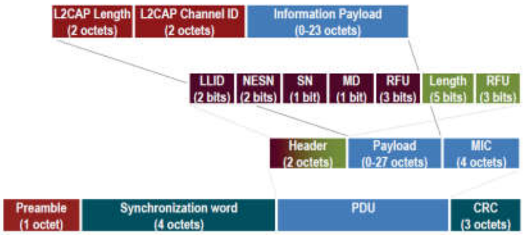
Figure 2.13: Data channel PDU header

  - ➢ LLID: Logical Link Identify

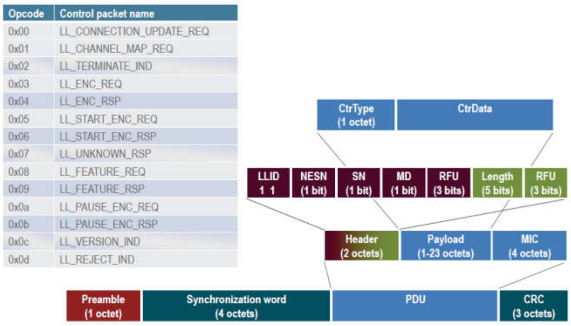


LLID	The LLID indicates whether the packet is an LL Data PDU or an LL Control PDU. 00b = Reserved 01b = LL Data PDU: Continuation fragment of an L2CAP message, or an Empty PDU. 10b = LL Data PDU: Start of an L2CAP message or a complete L2CAP message with no fragmentation. 11b = LL Control PDU

  - ➢ L2CAP:

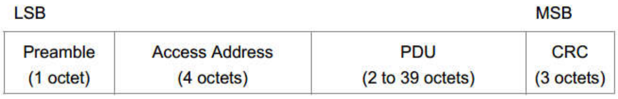


  - ➢ Link Control



Opcode	Control packet name
0x00	LL_CONNECTION_UPDATE_REQ
0x01	LL_CHANNEL_MAP_REQ
0x02	LL_TERMINATE_IND
0x03	LL_ENC_REQ
0x04	LL_ENC_RSP
0x05	LL_START_ENC_REQ
0x06	LL_START_ENC_RSP
0x07	LL_UNKNOWN_RSP
0x08	LL_FEATURE_REQ
0x09	LL_FEATURE_RSP
0x0a	LL_PAUSE_ENC_REQ
0x0b	LL_PAUSE_ENC_RSP
0x0c	LL_VERSION_IND
0x0d	LL_REJECT_IND

3.3 Packet Format Description



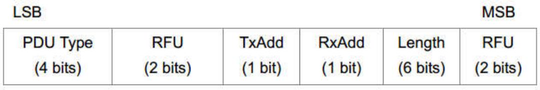
LSB			MSB
Preamble (1 octet)	Access Address (4 octets)	PDU (2 to 39 octets)	CRC (3 octets)

Figure 2.1: Link Layer packet format

- Used for Advertising and Data Channel Packets
- Preamble (0x55, 0xAA)
  - All Link Layer packets have an eight bit preamble. The preamble is used in the receiver to perform frequency synchronization, symbol timing estimation, and Automatic Gain Control (AGC) training
  - Preamble is either 10101010b or 01010101b, depending on the LSB of the Access Address. If the LSB of the Access is 1, the preamble shall be 01010101b; otherwise the preamble shall be 10101010b.
- The Access Address
  - Access Address for all advertising channel packets shall be 0x8E89BED6
  - The Access Address in data channel packets shall be different for each Link Layer connection between any two devices
- Packet Data Unit (Defined based upon packet types)

3.3.1 Advertising Packet Data Unit

PDU Type($b_3b_2b_1b_0$)	Packet	Description
0000	ADV_IND	connectable undirected advertising event
0001	ADV_DIRECT_IND	connectable directed advertising event
0010	ADV_NONCONN_IND	non-connectable undirected advertising event
0011	SCAN_REQ	Scan request sent by master or client device
0100	SCAN_RSP	Scan response sent by slave(server_ device
0101	CONNECT_REQ	Connect request send by master(client) device
0110	ADV_SCAN_IND	scannable undirected advertising event
0111~1111	Reserved	



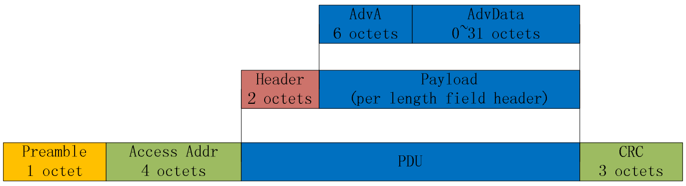
LSB					MSB
PDU Type (4 bits)	RFU (2 bits)	TxAdd (1 bit)	RxAdd (1 bit)	Length (6 bits)	RFU (2 bits)

Figure 2.3: Advertising channel PDU Header

- PDU Type: Advertising Packet Type
- RFU: Reversed for Future Use
- TxAdd: indicates whether the advertiser's address in the AdvA field is public (TxAdd = 0) or random (TxAdd = 1).
- RxAdd: indicates whether the initiator's address in the InitA field is public (RxAdd = 0) or random (RxAdd = 1).
- Length is the length of packet body (not include header)

3.3.1.1 Advertising PDU (Indirect Advertising)

PDU Type $(b_3b_2b_1b_0)$	Packet	Description
0000	ADV_IND	connectable undirected advertising event
0010	ADV_NONCONN_IND	non-connectable undirected advertising event
0110	ADV_SCAN_IND	scannable undirected advertising event



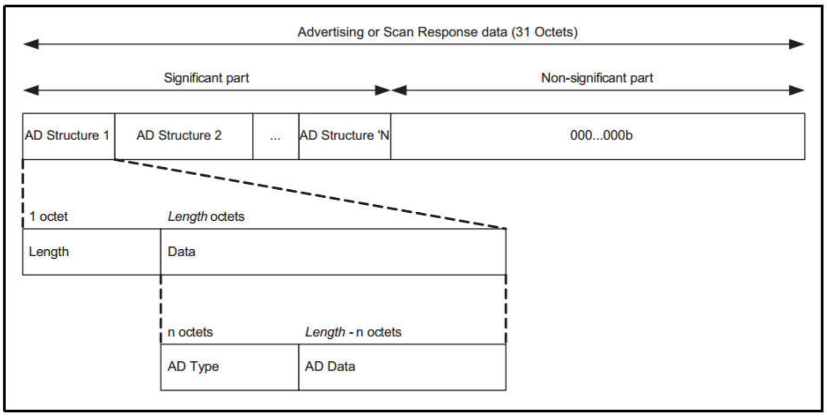
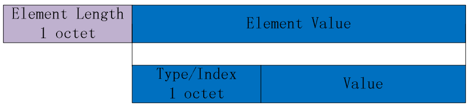
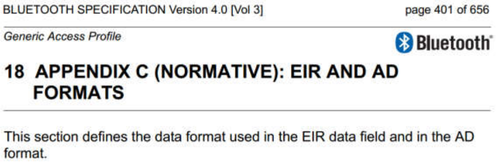


Figure 11.1: Advertising and Scan Response data format

- AdvA: Advertising Address (according to the TxAddr field in header)
- AdvData: Advertising Data
  - Format of AdvData:



    - The first byte represents the number of bytes left to the end of the AD structure. This allows a receiver of this structure to know when it ends and when a new AD structure starts.
    - The second byte is the ID of an AD structure type. https://www.bluetooth.com/specifications/assigned-numbers/generic-access-profile
    - The rest of the bytes are data that is structured in a predefined way depending on what AD type was defined by the previous byte.
  - Format of AdvElement:



BLUETOOTH SPECIFICATION Version 4.0 [Vol 3] page 401 of 656

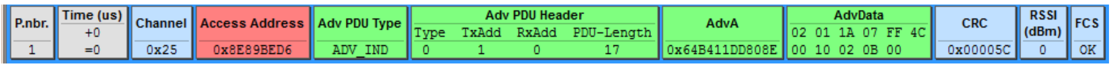
Generic Access Profile

Bluetooth

18 APPENDIX C (NORMATIVE): EIR AND AD FORMATS

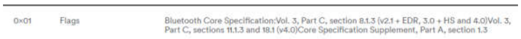
This section defines the data format used in the EIR data field and in the AD format.

- Instance:



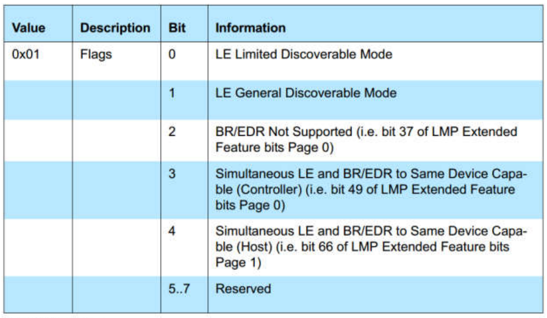
P.nbr.	Time (us)	Channel	Access Address	Adv PDU Type	Adv PDU Header				AdvA	AdvData	CRC	RSSI (dBm)	FCS
	+0				Type	TxAdd	RxAdd	PDU-Length		02 01 1A 07 FF 4C			
1	=0	0x25	0x8E89BED6	ADV_IND	0	1	0	17	0x64B411DD808E	00 10 02 0B 00	0x00005C	0	OK

- ✧ 02 01 1A:
  - ✓ 02: length
  - ✓ 01: Data type for Flags,



0x01	Flags	Bluetooth Core Specification:Vol. 3, Part C, section 8.1.3 (v2.1 + EDR, 3.0 + HS and 4.0)Vol. 3, Part C, sections 11.1.3 and 18.1 (v4.0)Core Specification Supplement, Part A, section 1.3

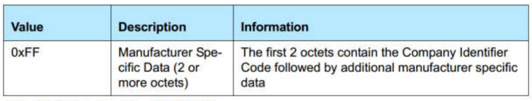
  - ✓ 1A (0001 1010b): LE General Discoverable Mode and Simultaneous LE and BR/EDR to sample device capable.



Value	Description	Bit	Information
0x01	Flags	0	LE Limited Discoverable Mode
		1	LE General Discoverable Mode
		2	BR/EDR Not Supported (i.e. bit 37 of LMP Extended Feature bits Page 0)
		3	Simultaneous LE and BR/EDR to Same Device Capable (Controller) (i.e. bit 49 of LMP Extended Feature bits Page 0)
		4	Simultaneous LE and BR/EDR to Same Device Capable (Host) (i.e. bit 66 of LMP Extended Feature bits Page 1)
		5..7	Reserved

Table 18.1: Flags

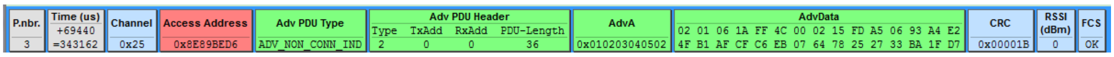
- ✧ 07 FF 4C 00 10 02 0B 00:
  - ✓ 07: length
  - ✓ FF: Manufacturer Specific Data



Value	Description	Information
0xFF	Manufacturer Specific Data (2 or more octets)	The first 2 octets contain the Company Identifier Code followed by additional manufacturer specific data

Table 18.11: Manufacturer Specific Data

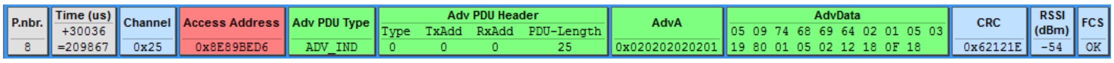
  - ✓ 4C 00 10 02 0B 00:
    - 4C 00 (0x004C): is Company Idenfier of Apple https://www.bluetooth.com/specifications/assigned-numbers/company-identifiers
    - Telink Semiconductor Co.ltd is 529 (0x0211)
    - 10 02 0B 00 is the data of iBeacon



P.nbr.	Time (us)	Channel	Access Address	Adv PDU Type	Adv PDU Header				AdvA	AdvData	CRC	RSSI (dBm)	FCS
	+69440				Type	TxAdd	RxAdd	PDU-Length		02 01 06 1A FF 4C 00 02 15 FD A5 06 93 A4 E2			
3	=343162	0x25	0x8E89BED6	ADV_NON_CONN_IND	2	0	0	36	0x010203040502	4F B1 AF CF C6 EB 07 64 78 25 27 33 BA 1F D7	0x00001B	0	OK

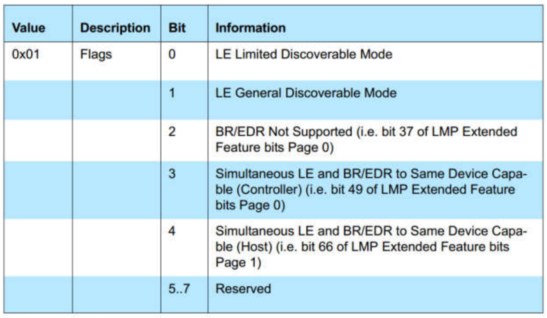
- ✧ 02 01 06
  - ✓ 06 (0000 0110b): LE General Discoverable Mode and Simultaneous LE and BR/EDR not support.
- ✧ 1A FF 4C 00 02 15 FD A5 06 93 A4 E2 4F B1 AF CF C6 EB 07 64 78 25 27 33 BA 1F D7
  - ✓ 0x1A (26): length
  - ✓ FF: Manufacturer Specific Data
  - ✓ 4C 00: Apple Company Idenfier
  - ✓ 02: iBeacon index
  - ✓ 0x15 (21): length of iBeacon
  - ✓ UUID: 0xFDA50693-A4E2-4FB1-AFCF-C6EB07647825
  - ✓ Version of Major: 0x2725
  - ✓ Version of Minor: 0xBA1F
  - ✓ RSSI: D7 (-41)

https://glimwormbeacons.com/learn/what-makes-an-ibeacon-an-ibeacon/



P.nbr.	Time (us)	Channel	Access Address	Adv PDU Type	Adv PDU Header				AdvA	AdvData	CRC	RSSI (dBm)	FCS
					Type	TxAdd	RxAdd	PDU-Length					
8	+30036 =209867	0x25	0x8E89BED6	ADV_IND	0	0	0	25	0x020202020201	05 09 74 68 69 64 02 01 05 03 19 80 01 05 02 12 18 0F 18	0x62121E	-54	OK

- ✧ 05 09 74 68 69 64
  - ✓ 09: Complete Local Name
  - ✓ 74 68 69 64: ‘t’, ‘h’, ‘i’, ‘d’, local name is “thid”
- ✧ 02 01 05
  - ✓ 05 (0000 0101b): Limited Discoverable Mode and BR/EDR not supported.



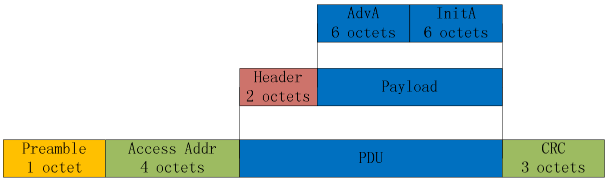
Value	Description	Bit	Information
0x01	Flags	0	LE Limited Discoverable Mode
		1	LE General Discoverable Mode
		2	BR/EDR Not Supported (i.e. bit 37 of LMP Extended Feature bits Page 0)
		3	Simultaneous LE and BR/EDR to Same Device Capable (Controller) (i.e. bit 49 of LMP Extended Feature bits Page 0)
		4	Simultaneous LE and BR/EDR to Same Device Capable (Host) (i.e. bit 66 of LMP Extended Feature bits Page 1)
		5..7	Reserved

Table 18.1: Flags

- ✧ 03 19 80 01
  - ✓ 19: Appearance
  - ✓ 80 01 (0x0180 or 384d): Generic Remote Control https://www.bluetooth.com/specifications/gatt/viewer?attributeXmlFile=org.bluetooth.characteristic.gap.appearance.xml
- ✧ 05 02 12 18 0F 18
  - ✓ 02: Incomplete List of 16-bit Service Class UUIDs
  - ✓ 12 18 0F 18 (0x180F1812): UUID

3.3.1.2 Advertising PDU (Direct Advertising)

PDU Type ($b_3b_2b_1b_0$)	Packet	Description
0001	ADV_DIRECT_IND	connectable directed advertising event



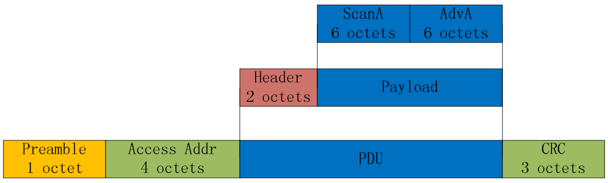
- AdvA: Advertising Address (according to the TxAddr field in header)
- InitA: Init Address (according to the RxAddr field in header)

3.3.1.3 SCAN PDU

PDU Type ($b_3b_2b_1b_0$)	Packet	Description
0011	SCAN_REQ	Scan request sent by master or client device

0100	SCAN_RSP	Scan response sent by slave or server device

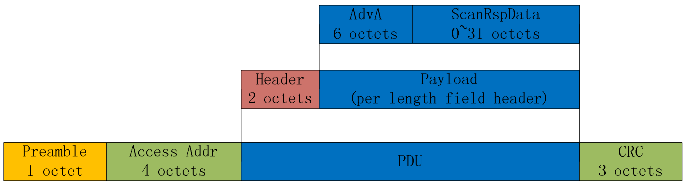
- SCAN_REQ:



✧ ScanA: Scanner Address (according to the TxAddr field in header)

✧ AdvA: Advertising recevie Address (according to the RxAddr field in header)

- SCAN_RSP:



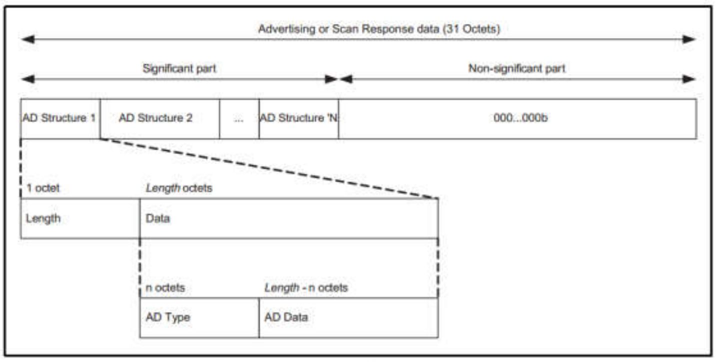
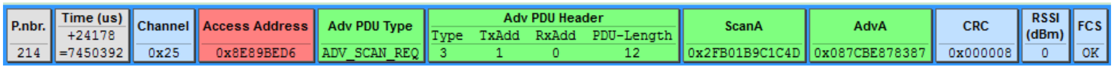


Figure 11.1: Advertising and Scan Response data format

- Instance:

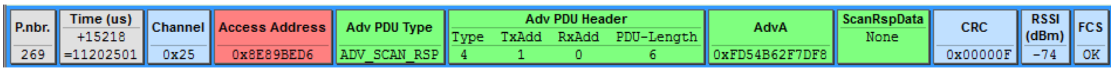
✧ SCAN REQ:



P.nbr.	Time (us) +24178	Channel	Access Address	Adv PDU Type	Adv PDU Header Type TxAdd RxAdd PDU-Length	ScanA	AdvA	CRC	RSSI (dBm)	FCS
214	=7450392	0x25	0x8E89BED6	ADV_SCAN_REQ	3 1 0 12	0x2FB01B9C1C4D	0x087CBE878387	0x000008	0	OK

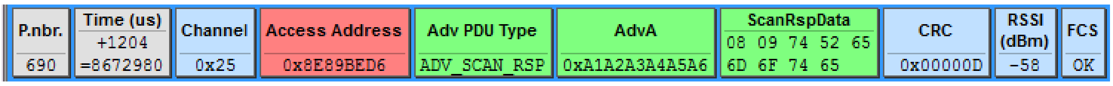
✧ SCAN RSP:

✧ SCAN RSP without data



P.nbr.	Time (us) +15218	Channel	Access Address	Adv PDU Type	Adv PDU Header Type TxAdd RxAdd PDU-Length	AdvA	ScanRspData None	CRC	RSSI (dBm)	FCS
269	=11202501	0x25	0x8E89BED6	ADV_SCAN_RSP	4 1 0 6	0xFD54B62F7DF8		0x00000F	-74	OK

✧ SCAN RSP with data



P.nbr.	Time (us) +1204	Channel	Access Address	Adv PDU Type	AdvA	ScanRspData 08 09 74 52 65	CRC	RSSI (dBm)	FCS
690	=8672980	0x25	0x8E89BED6	ADV_SCAN_RSP	0xA1A2A3A4A5A6	6D 6F 74 65	0x00000D	-58	OK

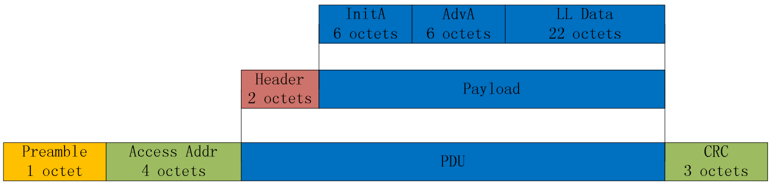
✓ 08: length

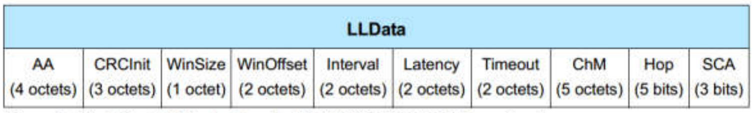
✓ 09: Complete Local Name
https://www.bluetooth.com/specifications/assigned-numbers/generic-access-profile

✓ 74 52 65 6D 6F 74 65: ‘t’, ‘R’, ‘e’, ‘m’, ‘o’, ‘t’, ‘e’, the local name is “tRemote”.

3.3.1.4 Init PDU (CONNECT)

PDU Type ($b_3b_2b_1b_0$)	Packet	Description
0101	CONNECT_REQ	Connect request send by master or client device

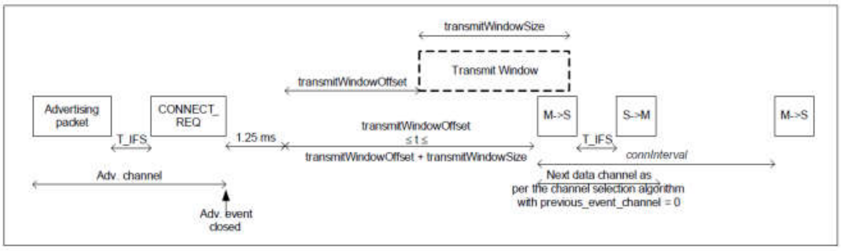




LLData									
AA (4 octets)	CRCInit (3 octets)	WinSize (1 octet)	WinOffset (2 octets)	Interval (2 octets)	Latency (2 octets)	Timeout (2 octets)	ChM (5 octets)	Hop (5 bits)	SCA (3 bits)

Figure 2.11: LLData field structure in CONNECT_REQ PDU's payload

- **The LLData consists of 10 fields:**
  - ✧ The AA field shall contain the Link Layer connection's Access Address determined by the Link Layer following the rules specified in Section 2.1.2.
  - ✧ The CRCInit field shall contain the initialization value for the CRC calculation for the Link Layer connection, as defined in Section 3.1.1. It shall be a random value, generated by the Link Layer.
  - ✧ The WinSize field shall be set to indicate the transmitWindowSize value, as defined in Section 4.5.3 in the following manner: transmitWindowSize = WinSize * 1.25 ms.
  - ✧ The WinOffset field shall be set to indicate the transmitWindowOffset value, as defined in Section 4.5.3 in the following manner: transmitWindowOffset = WinOffset * 1.25 ms.
  - ✧ The Interval field shall be set to indicate the connInterval as defined in Section 4.5.1 in the following manner: connInterval = Interval * 1.25 ms.
  - ✧ The Latency field shall be set to indicate the connSlaveLatency value, as defined in Section 4.5.1 in the following manner: connSlaveLatency = Latency.



transmitWindowSize
Transmit Window
transmitWindowOffset
Advertising packet
CONNECT_REQ
T_IFS
1.25 ms
transmitWindowOffset ≤ t ≤ transmitWindowOffset + transmitWindowSize
M->S
T_IFS
S->M
connInterval
M->S
Adv. channel
Adv. event closed
Next data channel as per the channel selection algorithm with previous_event_channel = 0

Figure 4.11: Master's view on LL connection setup with a non-zero transmitWindowOffset

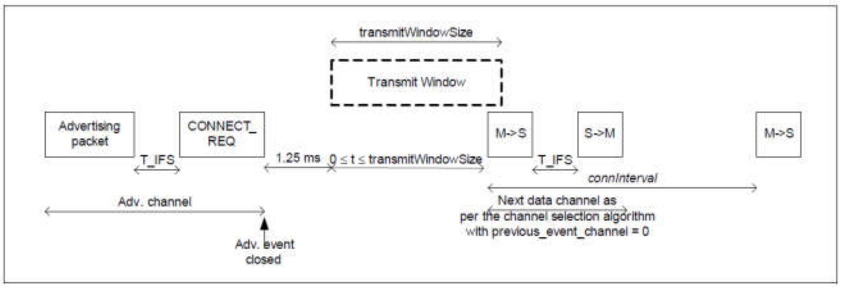
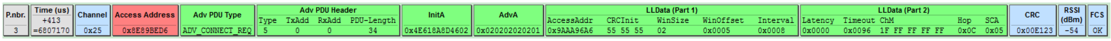


Figure 4.12: Master's view on LL connection setup with transmitWindowOffset set to zero

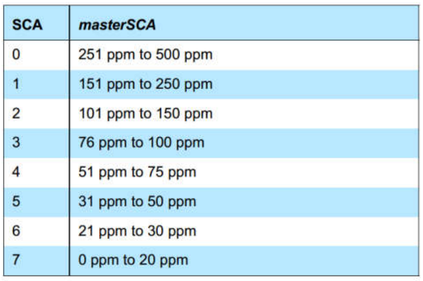
- ✧ The Timeout field shall be set to indicate the connSupervisionTimeout value, as defined in Section 4.5.2, connSupervisionTimeout is a parameter that defines the maximum time between two received Data Packet PDUs before the connection is considered lost. in the following manner: connSupervisionTimeout = Timeout * 10 ms.
- ✧ The ChM field shall contain the channel map indicating Used and Unused data channels. Every channel is represented with a bit positioned as per the data channel index as defined in Section 1.4.1. The LSB represents data channel index 0 and the bit in position 36 represents data channel index 36. A bit value of 0 indicates that the channel is Unused. A bit value of 1 indicates that the channel is used. The bits in positions 37, 38 and 39 are reserved for Future Use. Note: When mapping from RF Channels to data channel index, care should be taken to remember that there is a gap where the second advertising channel is placed.
- ✧ The Hop field shall be set to indicate the hopIncrement used in the data channel selection algorithm as defined in Section 4.5.8.2. It shall have a random value in the range of 5 to 16.
- ✧ The SCA field shall be set to indicate the masterSCA used to determine the worst case Master's sleep clock accuracy as defined in Section 4.2.2. The value of the SCA field shall be set as defined in Table 2.2.

- Instance:



P.nbr.	Time (us)	Channel	Access Address	Adv PDU Type	Adv PDU Header				InitA	AdvA	LLData (Part 1)					LLData (Part 2)					CRC	RSSI (dBm)	FCS
	+413				Type	TxAdd	RxAdd	PDU-Length			AccessAddr	CRCInit	WinSize	WinOffset	Interval	Latency	Timeout	ChM	Hop	SCA			
3	=6807170	0x25	0x8E89BED6	ADV_CONNECT_REQ	5	0	0	34	0x4E618A8D4602	0x020202020201	0x9AAA96A6	55 55 55	02	0x0005	0x0008	0x0000	0x0096	1F FF FF FF FF	0x0C	0x05	0x00E123	-54	OK

  - ✧ TxAdd/InitA: Random Address, 0x4E618A8D4602
  - ✧ RxAdd/AdvA: Random Address, 0x020202020201
  - ✧ Link Layer Access Address: 0x9AAA96A6
  - ✧ CRCInit: 55 55 55
  - ✧ transmitWindowSize = WinSize * 1.25ms = 0x02 * 1.25ms = 2.5ms
  - ✧ transmitWindowOffset = WinOffset * 1.25ms = 0x0005 * 1.25ms = 6.25ms
  - ✧ connInterval = Interval * 1.25 ms = 0x0008 * 1.25ms = 10ms
  - ✧ connSlaveLatency = Latency = 0x0000
  - ✧ connSupervisionTimeout = Timeout * 10 ms = 0x0096 * 10ms = 150 * 10ms = 1500ms.
  - ✧ ChM: 1F FF FF FF FF, all 37 channels will been used
  - ✧ Hop: 0x0C, Channel 12 will been used.
  - ✧ SCA: Sleep Clock Accuracy, 05, 31ppm ~ 50ppm, (31 ~ 50) * 12M / 10^6 = (372 ~ 600)Hz



SCA	*masterSCA*
0	251 ppm to 500 ppm
1	151 ppm to 250 ppm
2	101 ppm to 150 ppm
3	76 ppm to 100 ppm
4	51 ppm to 75 ppm
5	31 ppm to 50 ppm
6	21 ppm to 30 ppm
7	0 ppm to 20 ppm

Table 2.2: SCA field encoding

3.3.2 Data Packet Data Unit

The Data Channel PDU has a 16 bit header, a variable size payload, and may include a Message Integrity Check (MIC) field.

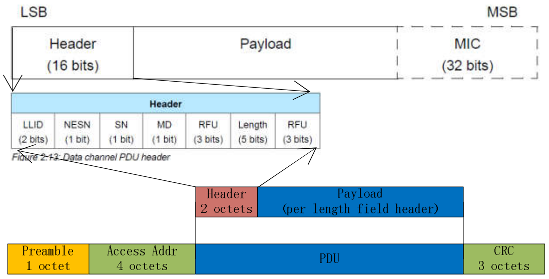
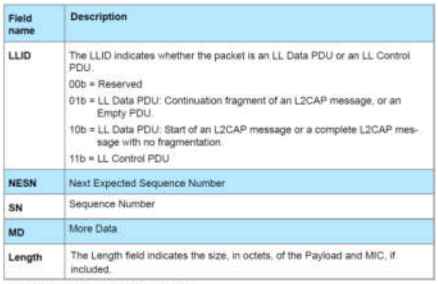


Figure 2.13: Data channel PDU header

The MIC field shall not be included in an un-encrypted Link Layer connection, or in an encrypted Link Layer connection with a data channel PDU with a zero length Payload.



Field name	Description
LLID	The LLID indicates whether the packet is an LL Data PDU or an LL Control PDU. 00b = Reserved 01b = LL Data PDU: Continuation fragment of an L2CAP message, or an Empty PDU. 10b = LL Data PDU: Start of an L2CAP message or a complete L2CAP message with no fragmentation. 11b = LL Control PDU
NESN	Next Expected Sequence Number
SN	Sequence Number
MD	More Data
Length	The Length field indicates the size, in octets, of the Payload and MIC, if included.

Table 2.3: Data channel PDU Header field

LLID code b_1b_0	Logical Link	Information
00	NA	undefined
01	ACL-U	Continuation fragment of an L2CAP message
10	ACL-U	Start of an L2CAP message or no fragmentation
11	ACL-C	LMP message

The master's Link Layer may send an Empty PDU to the slave to allow the slave to respond with any Data Channel PDU, including an Empty PDU.

3.3.2.1 LL Control PDU

An LL Control PDU is a Data Channel PDU that is used to control the Link Layer connection. An LL Control PDU shall not have the Length field set to 00000b. All LL Control PDUs have a fixed length, depending on the Opcode. The Opcode field identifies different types of LL Control PDU.

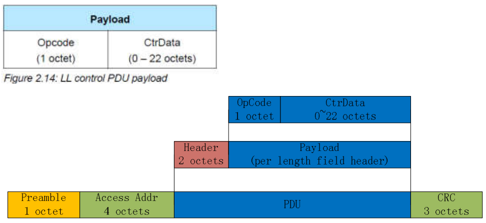
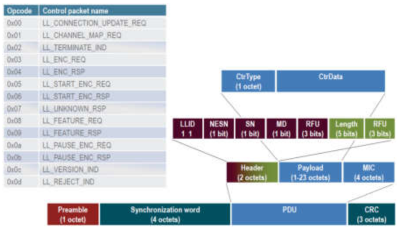


Figure 2.14: LL control PDU payload



Opcode	Control packet name
0x00	LL_CONNECTION_UPDATE_REQ
0x01	LL_CHANNEL_MAP_REQ
0x02	LL_TERMINATE_IND
0x03	LL_ENC_REQ
0x04	LL_ENC_RSP
0x05	LL_START_ENC_REQ
0x06	LL_START_ENC_RSP
0x07	LL_UNKNOWN_RSP
0x08	LL_FEATURE_REQ
0x09	LL_FEATURE_RSP
0x0a	LL_PAUSE_ENC_REQ
0x0b	LL_PAUSE_ENC_RSP
0x0c	LL_VERSION_IND
0x0d	LL_REJECT_IND

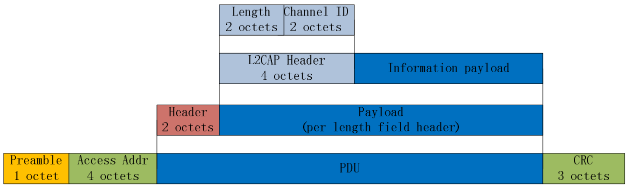
Opcode	Control PDU Name
0x00	LL_CONNECTION_UPDATE_REQ
0x01	LL_CHANNEL_MAP_REQ
0x02	LL_TERMINATE_IND
0x03	LL_ENC_REQ
0x04	LL_ENC_RSP
0x05	LL_START_ENC_REQ
0x06	LL_START_ENC_RSP
0x07	LL_UNKNOWN_RSP
0x08	LL_FEATURE_REQ
0x09	LL_FEATURE_RSP
0x0A	LL_PAUSE_ENC_REQ
0x0B	LL_PAUSE_ENC_RSP
0x0C	LL_VERSION_IND

Opcode	Control PDU Name
0x0D	LL_REJECT_IND
0x0E	LL_SLAVE_FEATURE_REQ
0x0F	LL_CONNECTION_PARAM_REQ
0x10	LL_CONNECTION_PARAM_RSP
0x11	LL_REJECT_IND_EXT
0x12	LL_PING_REQ
0x13	LL_PING_RSP
0x14	LL_LENGTH_REQ
0x15	LL_LENGTH_RSP
0x16-0xFF	Reserved for Future Use

Table 2.4: LL Control PDU Opcodes

3.3.2.2 LL Data PDU

L2CAP is packet-based but follows a communication model based on channels. A channel represents a data flow between L2CAP entities in remote devices. Channels may be connection-oriented or connectionless. Fixed channels other than the L2CAP connectionless channel (CID 0x0002) and the two L2CAP signaling channels (CIDs 0x0001 and 0x0005) are considered connection-oriented. All channels with dynamically assigned CIDs are connection-oriented.



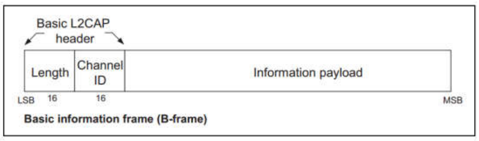


Figure 3.1: L2CAP PDU format in Basic L2CAP mode on connection-oriented channels (field sizes in bits)

- Length: 2 octets (16 bits)

  Length indicates the size of the information payload in octets, excluding the length of the L2CAP header. The length of an information payload can be up to 65535 octets. The Length field is used for recombination and serves as a simple integrity check of the recombined L2CAP packet on the receiving end.
- Channel ID: 2 octets

  The channel ID (CID) identifies the destination channel endpoint of the packet.
- Information payload: 0 to 65535 octets

  This contains the payload received from the upper layer protocol (outgoing packet), or delivered to the upper layer protocol (incoming packet). The MTU for channels with dynamically allocated CIDs is determined during channel configuration.