1 Overview

The Dongle Host Driver (DHD) offload feature refers to the capability of a system to offload the traffic classification, modification and forwarding to and from WLAN to the hardware accelerator, thus reducing the host CPU utilization.

The term DHD Offload is used within the scope of this document as follows:
- Dongle Offload and DHD Offload may be used interchangeably.
- Offload specifically means “Host side processing offload”.
- Dongle refers to Broadcom WLAN card with ‘PCIe Full Dongle Offload Support”, such as but not limited to BCM43602, BCM4366, BCM43684.
- The term does not apply to “NIC” mode WLAN drivers (including WLAN Dongles used in NIC mode).

DHD offload is currently supported on the following Runner Data Path (RDP) hardware accelerator based platforms:
- BCM63138
- BCM63148
- BCM63158
- BCM6838X
- BCM6858X
- BCM6856X
- BCM6836X
- BCM6846X
- BCM4908
- BCM62118

DHD Offload support is available when using Flow Cache module with Hardware acceleration, in either L3 or L2 modes. The mechanism of flow learning/provisioning by flow-cache into the RDP accelerator is outside the scope of this document.

This document does not intend to highlight the exact code execution path but is limited to briefly provide the high level packet/processing flow.

2 Build Instructions

DHD Offload feature is enabled by default for supported devices in WLAN RX direction. Enabling DHD offload in the TX direction requires dedicated memory reservation, this memory must be reserved at system boot time. Careful consideration is necessary when enabling this feature on low memory devices, see Memory Reservation.

To enable/disable the DHD offload feature before building:

1. Use the **make menuconfig** command at the Linux command prompt.
   ```
   $ make menuconfig
   ```
2. Navigate to menuconfig→Datapath→WLAN Acceleration
3. Select [ ] DHD Runner Acceleration Support
4. Save and exit

3 Memory Reservation

To enable DHD Offload WLAN TX functionality memory reservation must be done using the CFE boot loader parameters. Two CFE memory reservation parameters should be considered:

DHD memory allocation – memory allocation for the WLAN transmit rings

Packet buffering memory – memory allocation for the RDP packets buffering

Below is an example of CFE boot loader parameters for the BCM63138 reference board. Specified memory is reserved prior to Linux boot and will not be available for Linux usage.

```
Base MAC Address                    : 02:10:18:89:82:01
PSI Size (1-128) KBytes             : 48
Enable Backup PSI [0|1]             : 0
System Log Size (0-256) KBytes      : 0
Auxillary File System Size Percent: 0
MC memory allocation (MB)           : 4
TM memory allocation (MB)           : 44
DHD 0 memory allocation (MB)        : 7
DHD 1 memory allocation (MB)        : 4
DHD 2 memory allocation (MB)        : 0
```

3.1 DHD TX Rings Memory

Memory requirement for DHD Offload Transmit Rings depends upon the WLAN dongle radio type and max number of stations supported by the radio.

DHD 0/1/2 refers to the radio index that is assigned to the specific WLAN dongle during the PCIe scan (see Radio Number Assignment).

In the above example, the first WLAN dongle is allocated 7 MB and the second WLAN dongle is allocated 4 MB. When value of 0 MB is used, DHD TX offload is disabled for that radio.

specifies memory requirements for typical usage from 5.02L.05p1 release onwards.

WLAN Dongle	DHD Memory Reservation (Mb)		
	No Offload	Default	Flowring physical size=512
43684	0	11	9
4366	0	8	7
43602A3	0	6	5
43602A1	0	4	4

NOTE: Default has changed in 5.02L05p1 to support BCM43684 high traffic rates. It is possible to revert this change for BCM43684 with some TX performance impact, and with no performance impact for BCM4366/BCM43602. To view FlowRing_size, see Flow-Ring Profile User Interface .

Revert to overriding the default configuration in NVRAM using the below command, where *X* represents the radio index: `“nvram set dhdX_rnr_flowring_physize=512”`

NOTE: Reserved memory must be set to either “0 Mb” for no offload or size specified in the above table for full offload. Any memory size in between results in failure to load DHD and system may boot with wrong WLAN configuration.

NOTE: For BCM6838 different memory reservation size is required. Please open a support case to get the correct value.

3.2 Packets Buffering

When DHD offload is active, packet buffering for WLAN TX is handled by the RDP data path and therefore data path packet memory may need to be increased to accommodate the large buffering requirements towards the WLAN dongle.

For BCM63138/63148 and BCM6838X, the packet buffer is set by the CFE boot parameter “TM Memory allocation” as shown in the example above.

BCM63138/63148 recommended TM memory value with DHD offload is to 44 MB (default is 20 MB).

BCM96838X recommended TM memory value with DHD offload is 36 MB (default is 20 MB). ‘*MC memory allocation*’ should be set to 8 MB (default is 4 MB).

For the other platforms CFE boot parameter “*FPM_POOL memory allocation*” should be adjusted according to the type and number of offloaded radios and the maximal number stations:

- Applicable sizes are 16MB / 32MB/ 64 MB /128MB.
- For 4908/62118 applicable sizes are 32 MB / 64 MB.

4 DHD Interaction with Dongle

All communication with Dongle is via descriptor rings irrespective of offload or not. Dongle firmware remains the same in both cases and dongle is configured to work in offload/non-offload modes. The following type of descriptor rings are used:

- Packet Receive Path related rings
  - RxBuf Post Ring (RX_POST): DHD/RDP puts the descriptor for the packet receive buffer in this ring. Dongle uses these buffers to receive the packets from WLAN Clients and passes them back to DHD/RDP (using RxPkt Complete Ring). Only one RX_POST ring per Dongle is created.
  - RxPkt Complete Ring (RX_CMPL): Dongle puts the descriptors for received packets in this ring and signals the DHD/RDP to process the packet(s). Only one RX_CMPL ring per Dongle is created.
- Packet Transmit Path related rings
  - TxPkt Post Flow-ring (TX_POST/Flow-Ring): DHD/RDP puts the descriptor for packets that need to be transmitted towards Dongle (i.e., WLAN Station) and signals the Dongle to process the packet(s). Flow-rings are created by DHD per station based on priority. The size is configurable through the boot time profile. There are additional flow-rings for Broadcast/Multicast traffic for every SSID/BSS.
  - TxPkt Complete Ring (TX_CMPL): Dongle puts the descriptors related to transmitted packets in this ring and signals the DHD/RDP to indicate completion of transmission, so that DHD/RDP can free up the packet buffers. Only one TX_CMPL ring per Dongle is created.

DHD allocates all the rings at initialization time based on the configured sizes and provides the required information to Dongle.

4.1 Flow-Ring Profile User Interface

“dhdctl” provides IOCTLs to list and sets the flow ring profile for each radio. If no profile is stored in PSP, then DHD uses the default built-in profile ID 3.

To view the list of available profiles and the current active profile use the command shown below. The active profile of the radio is the id marked with an asterisk “*”.

```
# dhd -i wl[n] rnr_flowring_profile
[id] [ ac_bk ] [ ac_be ] [ ac_vi ] [ ac_vo ] [ bc_mc ]
  0     1024      2048     1024      0512      0512
  1     1024      2048     1024      0512      0512
  2     1024      2048     1024      0512      0512
 *3     1024      2048     1024      0512      0512
  7     2048      2048     2048      2048      2048
```

"n" is the radio index (0-2).

Flow-Ring profile specifies the total ring size (“physical flow-ring” + “backup queue”) for each WLAN access category. The size mentioned in the profile minus the physical flow-ring size specifies the flow-ring backup queue size in runner. The default size of physical flow-ring is set to 1024 entries for Best Effort access category, 512 for all other access categories and can be modified using nvram setting mentioned in section 3.1.

The setting can be applied using NVRAM `setting dhd[n]_rnr_flowring_profile`. All of the dhd_runner nvram settings requires set, commit, and reboot to take effective.

```
#nvram set dhd[n]_rnr_flowring_profile="[n] [ac_bk size] [ac_be size] [ac_vi size] [ac_vo size] [bc_mc size]"
#nvram commit
#reboot
```

Example: Select profile #7

```
#nvram set dhd0_rnr_flowring_profile=7
```

Example: Set parameters for profile #0 with samering size for BE, VI, VO access categories

```
#nvram set dhd0_rnr_flowring_profile=”0 1024 2048  2048  2048  512”
```

5 IPTV Multicast Handling

When DHD offload is active on a WLAN radio, multicast replications of packets directed to all stations joined the stream on this radio is fully handled by RDP, including multicast to unicast address translation.

When multiple clients join the same stream and the destination radio is managed by DHD, RDP only forwards one packet per SSID to DHD and the replication to all the clients is done by DHD.

NOTE: Multicast acceleration through RDP is only supported for WAN to LAN/WLAN traffic. LAN/WLAN to LAN/WLAN multicast acceleration is done by flow-cache in the Host/CPU only.
If the WMF (Wireless Multicast Forwarding) feature is NOT enabled for a given SSID/BSS, all multicast traffic towards clients on that SSID/BSS will go through the WFD acceleration path.

6 Inter/Intra BSS and Inter Radio Traffic Handling

Inter/Intra BSS and Inter Radio traffic will always be accelerated using one of the following acceleration paths.

- Non-offload builds
  - Flow-cache acceleration applies.
- Offload builds
  - RDP accelerates all the received traffic from Dongle as below:
    - RDP forwards the traffic through the WFD path if the destination is non offloaded DHD or NIC WLAN.
    - RDP directly forwards the traffic (without host involvement) if destination radio is offloaded by RDP.

– For all other scenarios Inter/Intra BSS and Inter Radio traffic are learned and accelerated by flow-cache.

7 Radio Number Assignment

During radio number assignment, dongle cards are scanned first. The radio number assignment starts from 0 and is incremented for each additional dongle card.

NOTE: NIC cards are scanned after dongle cards.

The following is the scan sequence:

- PCIe port #0
  – Port#0 … Port#N, if connected through a PCIe bridge
- PCIe port #1 (if present)
  – Port#0 … Port#N, if connected through a PCIe bridge
- PCIe port #2 (if present)
  – Port#0 … Port#N, if connected through a PCIe bridge

The WLAN interface number will be same as the DHD dongle radio number.

8 Packet Flow Diagrams

8.1 Packet Receive (Dongle to RDP)

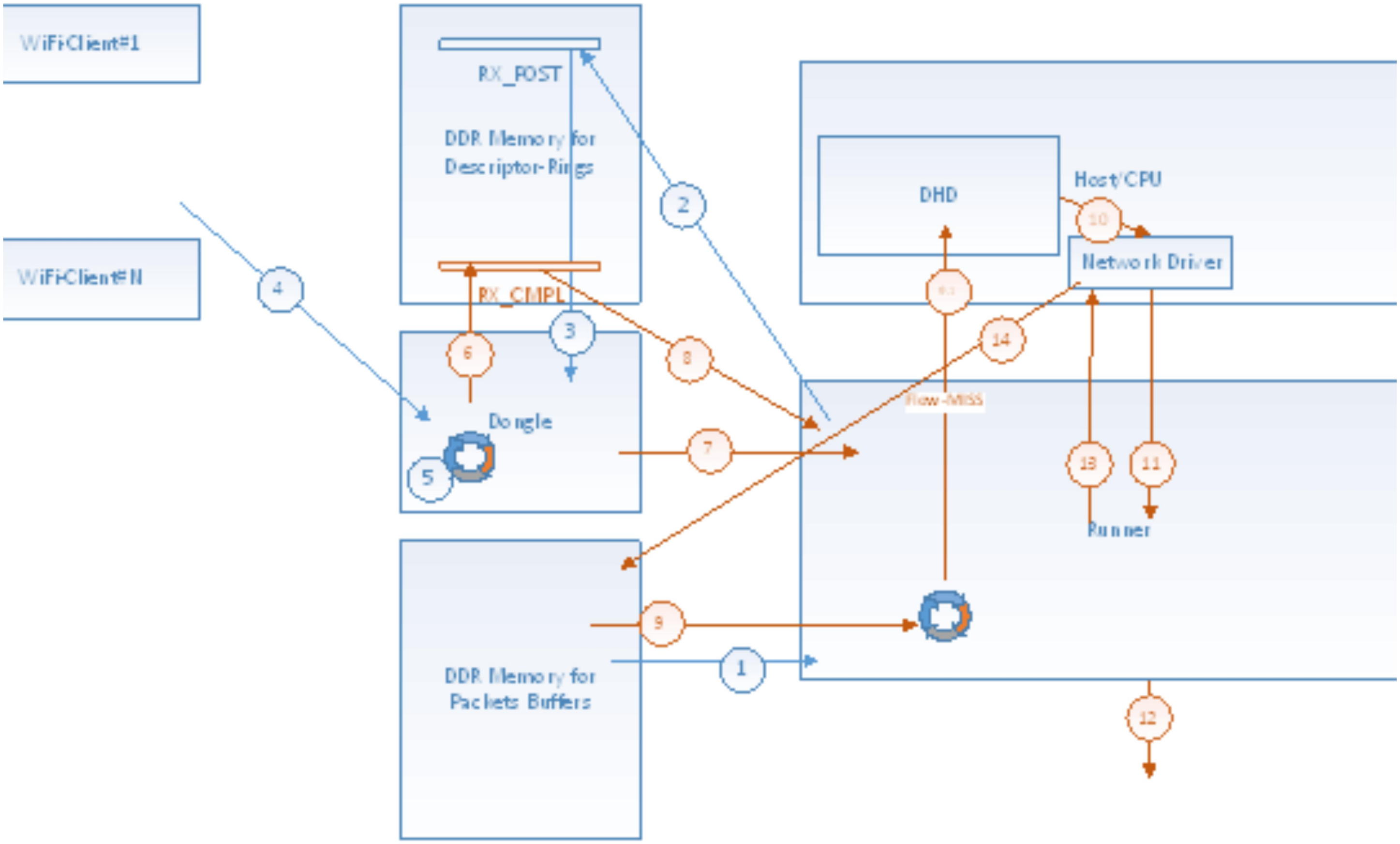
When DHD offload is enabled, all packets from Dongle are always received by the RDP (accelerated or not).

8.1.1 Receive Flow-Miss Case

The Flow-miss path (packet trapped to DHD for processing) will occur whenever RDP does not have the flow to process/forward the packet. This mechanism remain same as any other packet acceleration done by RDP for other interfaces (LAN/WAN).

The changes from the DHD receive path are highlighted in the text description to emphasis the fact that Dongle does not care if the packet is received by the RDP or DHD.

Figure 1: Packet Receive Flow-Miss Case



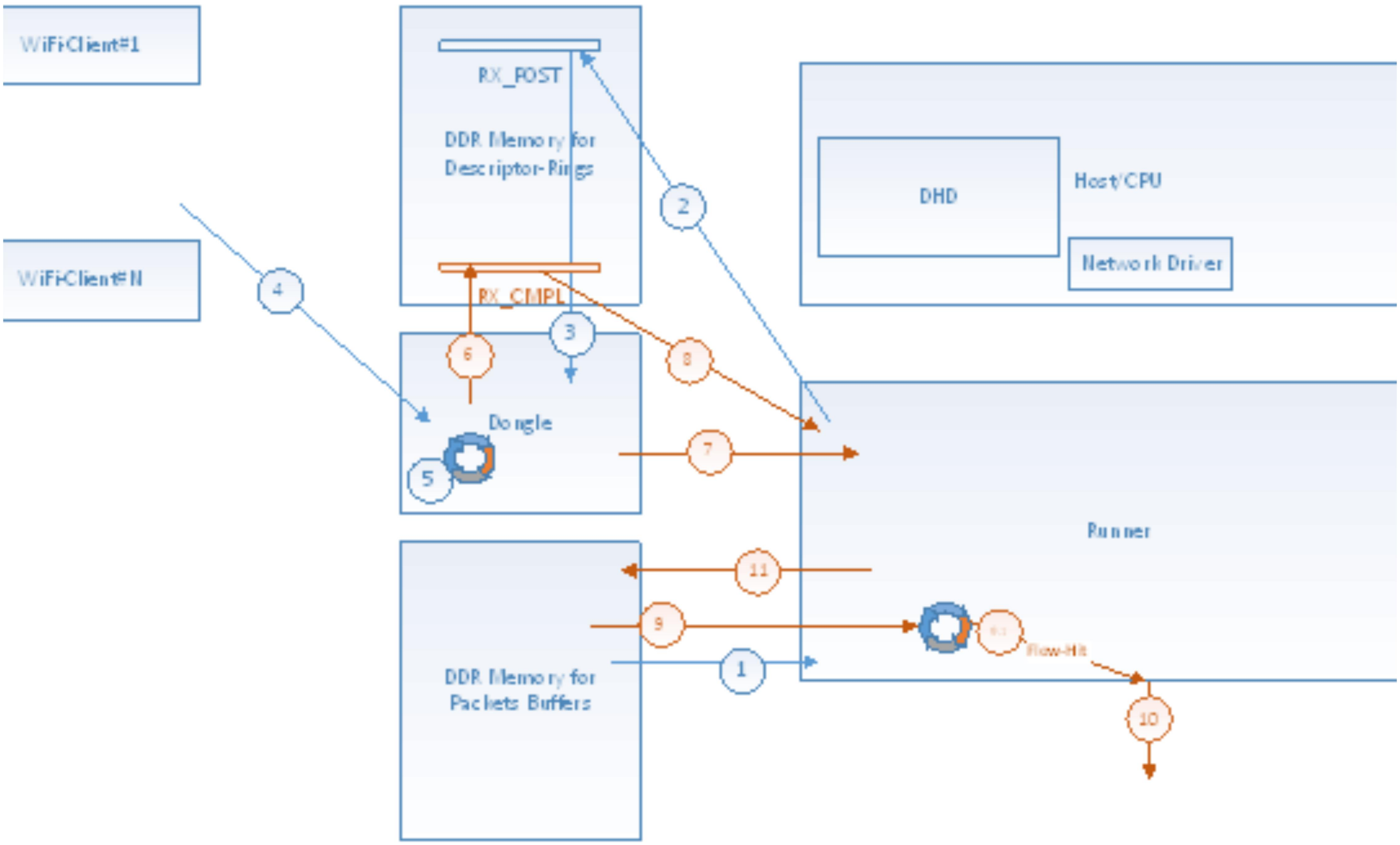
1. RDP allocates a packet buffer to receive the packet from Dongle.
2. RDP puts the Receive packet buffer information in RX_POST Descriptor ring.
3. Dongle fetches the receive packet buffer information from the RX_POST descriptor ring.
4. Dongle receives the packet from WiFi-Client into the packet buffer.
5. Dongle processes the packet.
6. Dongle puts the information about received packet in RX_CMPL descriptor ring.
7. Dongle notifies the RDP about the packet.
8. RDP fetches the descriptor from the RX_CMPL ring.
9. RDP fetches the packet from the packet buffers.
   – RDP has no flow for this packet → Flow-miss → thepacket is trapped to DHD.
10. DHD forwards the packet to Network Driver (either through Linux or flow-cache).
11. Network Driver gives the packet to RDP for transmission.
12. RDP transmits the packet towards LAN/WAN.
13. RDP sends the packet buffer back to Network Driver to free.
14. Network Driver frees up the buffer back to Packet buffer memory.

8.1.2 Receive Flow-hit Case

The Flow-hit path will occur whenever RDP has the flow to process/forward the packet. This mechanism remains the same as any other packet acceleration done by RDP for other interfaces (LAN/WAN).

Flows are pushed to RDP by flow-cache the same way as done for any other interface. The changes from the “Flow-miss” receive path are highlighted in the text description below.

Figure 2: Packet Receive Flow-hit Case



1. RDP allocates a packet buffer to receive the packet from Dongle.
2. RDP puts the Receive packet buffer information in RX_POST Descriptor ring.
3. Dongle fetches the receive packet buffer information from the RX_POST descriptor ring.
4. Dongle receives the packet from WiFi-Client into the packet buffer.
5. Dongle processes the packet.
6. Dongle puts the information about received packet in RX_CMPL descriptor ring.
7. Dongle notifies the RDP about packet.
8. RDP fetches the descriptor from the RX_CMPL ring.
9. RDP fetches the packet from the packet buffers.
   – RDP has flow for this packet → Flow-hit → the packet is forwarded to the egress interface.

10. RDP transmits the packet towards LAN/WAN.
11. RDP frees up the buffer back to Packet buffer memory.

8.2 Packet Transmission (RDP/DHD to Dongle)

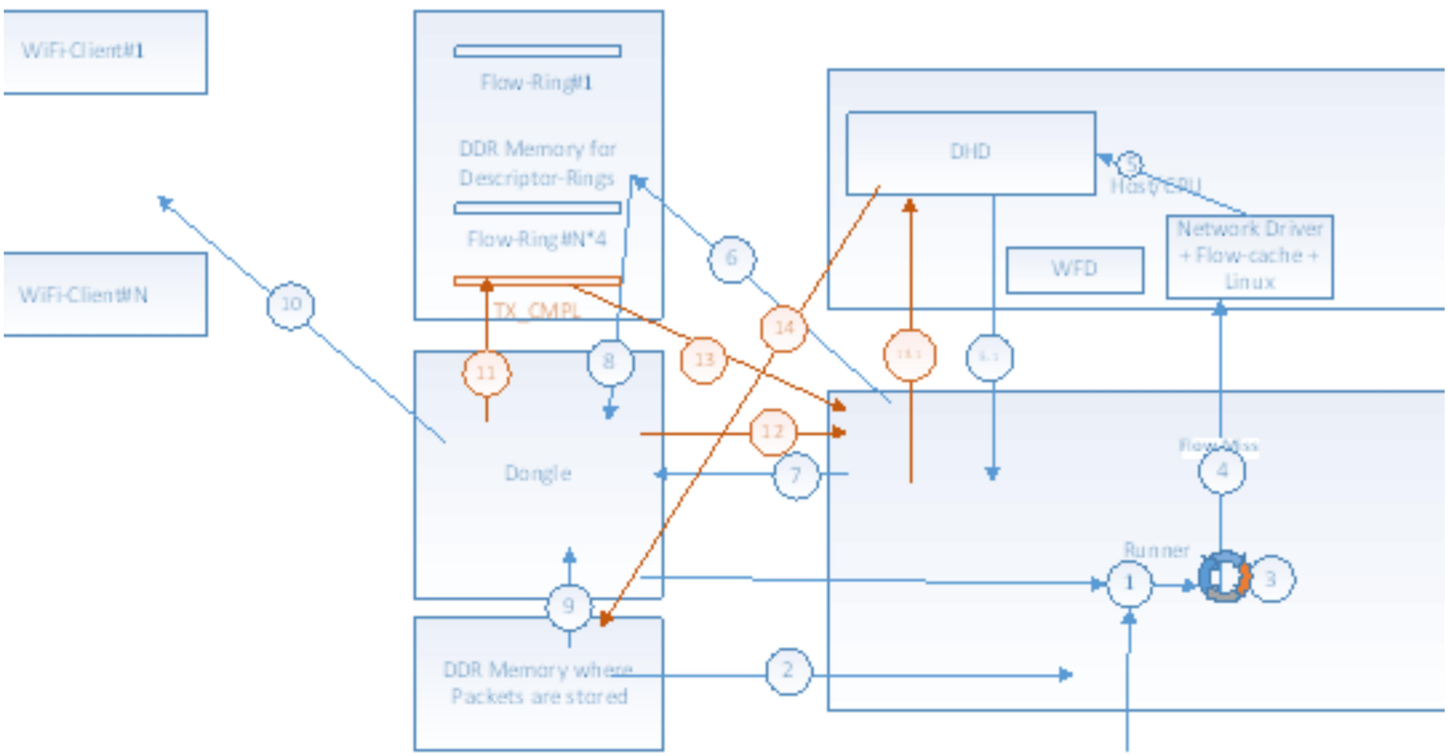
This case illustrates the data path flow for two scenarios, flow-miss and flow-hit.

8.2.1 Flow-Miss Case

Figure 3: The differences between the offload and the non-offload flow-miss case are highlighted in

.

Figure 3: Packet Transmit Flow-miss Case



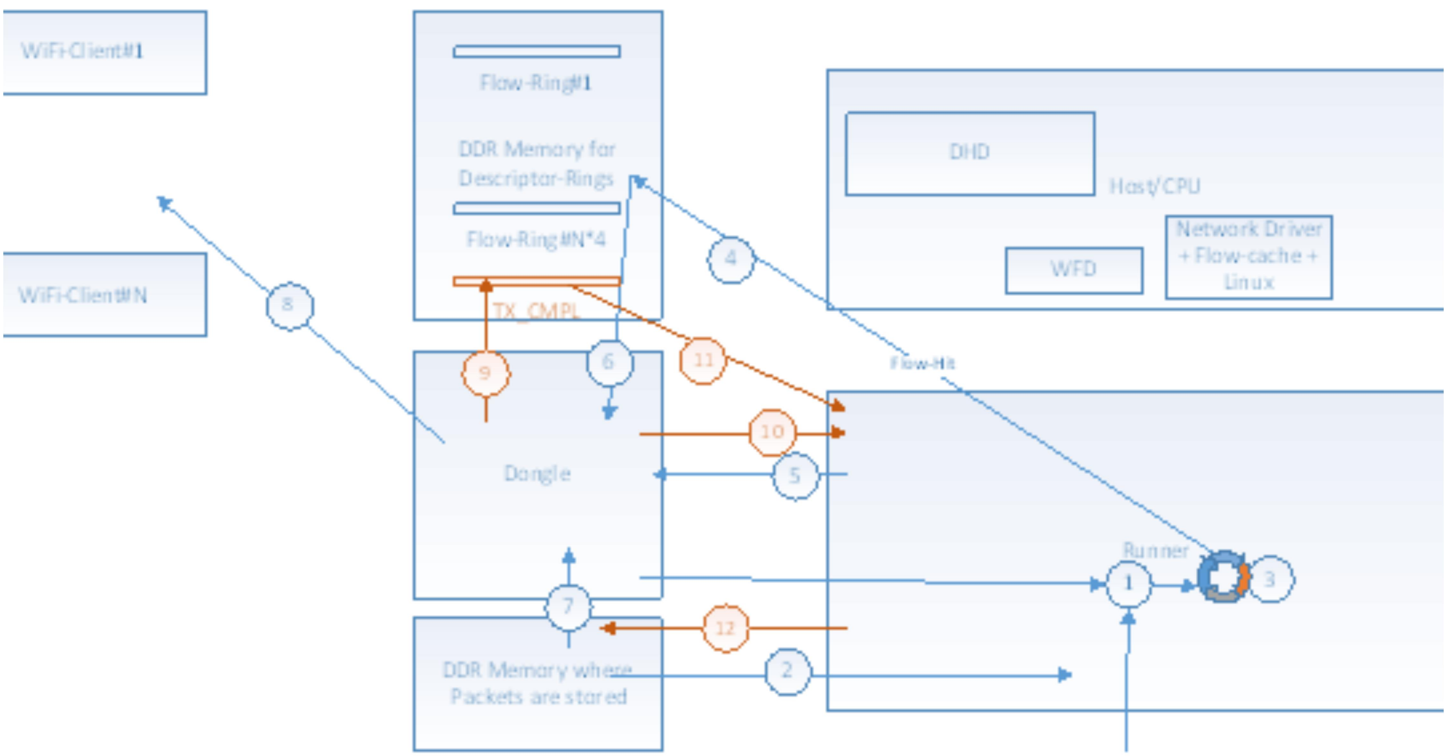
1. Packet is received by RDP from LAN/WAN port or WLAN-Dongle.
2. Packet is received in the DDR memory buffer.
3. RDP processes the packets and makes the forwarding decision (flow-miss).
4. RDP sends the packet to the Host CPU Network driver.
5. Network driver receives the packet and forwards it to the DHD Driver (either through Linux Kernel, or Flow-cache).
   – DHD determines that the flow-ring is managed by RDP, and forwards the packet to the RDP for transmission.
6. RDP updates the TX_POST descriptor in the flow-ring with required information.

7. RDP notifies the Dongle about the packet.
8. Dongle reads the TX_POST descriptor from flow-ring and gets the required information.
9. Dongle grabs the packets from DDR.
10. Dongle transmits the packets towards Wi-Fi-Client.
11. Dongle updates the descriptor with transmitted packet buffer information in TX_CMPL Ring.
12. Dongle notifies the RDP about the TX_CMPL ring update.
13. RDP takes the packet buffer information from the TX_CMPL ring.
14. RDP frees the packet buffer back to the packet pool (from wherever it came from).

8.2.2 Flow-Hit Case

The differences between the offload and the non-offload flow-hit case are highlighted in Packet Transmit Flow-Hit Case.

Figure 4: Packet Transmit Flow-Hit Case



1. A packet is received by RDP from LAN/WAN port or WLAN-Dongle.
2. The packet is stored in the DDR memory.
3. RDP processes the packet and makes the forwarding decision (flow-hit).

4. RDP updates the TX_POST descriptor in the flow-ring with required information.
5. RDP notifies the Dongle about the packet.
6. Dongle reads the TX_POST descriptor from flow-ring and gets the required information.
7. Dongle grabs the packets from DDR.
8. Dongle transmits the packets towards WiFi-Client.
9. Dongle updates the descriptor with transmitted packet buffer information in TX_CMPL Ring.
10. Dongle notifies RDP about the TX_CMPL ring update.
11. RDP takes the packet buffer information from the TX_CMPL ring.
12. RDP frees the packet buffer back to the packet pool (from wherever it came from).

9 DHD Offload Debugging

While debugging any issues related to the DHD Offload, the user should look for the following information:

1. Check the amount of memory allocated through the boot loader. See Memory Reservation for details.
2. Make sure that the correct memory is allocated during system boot (as configured in the boot loader). Look for these outputs during system boot to find out the actual memory allocated.

```
creating a MT_MEMORY_NONCACHED device at physical address of 0x0d200000 to virtual address at
0xcd200000 with size of 0x2c00000 byte for RDPA tm
creating a MT_MEMORY_NONCACHED device at physical address of 0x0c000000 to virtual address at
0xcc000000 with size of 0xe00000 byte for DHD dhd0
creating a MT_MEMORY_NONCACHED device at physical address of 0x0b800000 to virtual address at
0xcb800000 with size of 0x700000 byte for DHD dhd1
```

3. Get the output from the following commands:
   – “dhdctl –i wl0 dump”
   – “dhdctl –i wl1 dump”

```
DHD Runner:
  Status     : radio#  0 tx_Offl 1  rx_Offl 1
  Profile    : prfl_id 3 id_valu  1024   2048   1024   0512   0512
  h2r_notif  : tx_post 74 rx_cmpl 5 tx_cmpl 22
  r2h_req    : rx_cmpl 11 tx_cmpl 0 wk_dngl 85

CtrlPost: RD 37 WR 37
CtrlCpl:  RD 85 WR 85
RxPost:   RBP 0 RD 576 WR 10
RxCpl:    RD 11 WR 11
TxCpl:    RD 73 WR 74
active_tx_count 0  pktidmap_avail -1  rxbufpost 0
dhd cumm_ctr 0
Num: HW Flow If Prio :Dest_MacAddress: Qlen CLen L2CLen Pkts  Overflows  RD  WR  Acked tossed noack
  0.  1   2  0    0
33:33:00:00:00:16    0    0  0       0       0       69   69    NA     NA    NA
  1.  1  202  0   0
ac:9e:17:45:a1:ac    0    0  0       0       0        5   5    NA     NA    NA
```

The code above the configuration information as follows:

- The Flow-Ring Profile used.
- RX_POST, RX_CMPL, TX_CMPL, and TX_POST ring current Read/RD and Write/WR pointers.
- Provides information about the Flow-Rings:
  - Flow-Ring-ID “Flow” (e.g., 202)
  - Is the flow-ring managed by hardware (i.e., RDP) or software (i.e., DHD) – shown in the “HW” column.
  - Priority (or Access Category) of the flow-ring – shown in the “Prio” column.
  - MAC Address of WLAN Client using that flow-ring – shown in the “Dest MacAddress: column.

4. Dump RDP DHD offload statistics using command “bs /b/e dhd_helper <radio_num>”

```
ssid_tx_dropped_packets[0] = 2700       // drops per SSID, due to full TX rings
dhd_stat.dhd_tx_fpm_used = 1570         // current number of RDP buffers used by TX rings

dhd_stat.dhd_tx_fr_ac_be_full = 10211    // total drops per access category
dhd_stat.dhd_tx_post_packets = 6927858
dhd_stat.dhd_tx_complete_packets = 6927960
dhd_stat.dhd_rx_complete_packets = 3078349
```

5. If the issue is related to unicast acceleration, dump the RDP unicast flows.
6. If the issue is related to multicast acceleration, dump the running multicast flows along with WLAN multicast flow information using command “bs /b/e wlan_mcast”

9.1 Enable RX/TX Offload

Receive and transmit offload can be enabled/ disabled as follows:

Option 1 – TX enable/disable from CFE

- Update the DHD 0/1/2 memory allocation. See Memory Reservation“.
- Allocated value of ‘0’ disables the TX offload.
- Boot to CFE and press 'b'.

Option 2 – TX/RX enable/disable form NVRAM:

- nvram set dhd[n]_rnr_rxoffl=[0/1]   1 - enable, 0 - disable
- nvram set dhd[n]_rnr_txoffl=[0/1]
- nvram commit
- reboot