# DPAA2 Ethernet driver

This file provides documentation for the Freescale DPAA2 Ethernet driver.

## Supported Platforms

This driver provides networking support for Freescale DPAA2 SoCs, e.g. LS2080A, LS2088A, LS1088A.

## Architecture Overview

Unlike regular NICs, in the DPAA2 architecture there is no single hardware block representing network interfaces; instead, several separate hardware resources concur to provide the networking functionality:

- network interfaces
- queues, channels
- buffer pools
- MAC/PHY

All hardware resources are allocated and configured through the Management Complex (MC) portals. MC abstracts most of these resources as DPAA2 objects and exposes ABIs through which they can be configured and controlled. A few hardware resources, like queues, do not have a corresponding MC object and are treated as internal resources of other objects.

For a more detailed description of the DPAA2 architecture and its object abstractions see *Documentation/networking/device_drivers/ethernet/freescale/dpaa2/overview.rst*.

Each Linux net device is built on top of a Datapath Network Interface (DPNI) object and uses Buffer Pools (DPBPs), I/O Portals (DPIOs) and Concentrators (DPCONs).

Configuration interface:

```
                -----------------------
               | DPAA2 Ethernet Driver |
                -----------------------
                    .      .      .
                    .      .      .
            . . . . .      .      . . . . . .
            .              .                .
            .              .                .
        ----------     ----------      -----------
       | DPBP API |   | DPNI API |    | DPCON API |
        ----------     ----------      -----------
            .              .                .             software
   =======  .  ==========  .  ============  .  ===================
            .              .                .             hardware
        ------------------------------------------
       |            MC hardware portals           |
        ------------------------------------------
            .              .                .
            .              .                .
         ------         ------            -------
        | DPBP |       | DPNI |          | DPCON |
         ------         ------            -------
```

The DPNIs are network interfaces without a direct one-on-one mapping to PHYs. DPBPs represent hardware buffer pools. Packet I/O is performed in the context of DPCON objects, using DPIO portals for managing and communicating with the hardware resources.

Datapath (I/O) interface:

```
           -----------------------------------------------
          |           DPAA2 Ethernet Driver               |
           -----------------------------------------------
            |          ^        ^         |            |
            |          |        |         |            |
     enqueue|   dequeue|   data |  dequeue|       seed |
      (Tx)  | (Rx, TxC)|  avail.|  request|     buffers|
            |          |  notify|         |            |
            |          |        |         |            |
            V          |        |         V            V
           -----------------------------------------------
          |                 DPIO Driver                   |
           -----------------------------------------------
            |          |        |         |            |          software
```

```
             |          |        |         |            |  ================
             |          |        |         |            |          hardware
           -----------------------------------------------
          |                I/O hardware portals             |
           -----------------------------------------------
             |          ^        ^         |            |
             |          |        |         |            |
             |          |        |         V            |
             V          |    ================           V
          ----------------------             |      -------------
   queues  ----------------------            |     | Buffer pool |
            ----------------------           |      -------------
                     =======================
                                  Channel
```

Datapath I/O (DPIO) portals provide enqueue and dequeue services, data availability notifications and buffer pool management. DPIOs are shared between all DPAA2 objects (and implicitly all DPAA2 kernel drivers) that work with data frames, but must be affine to the CPUs for the purpose of traffic distribution.

Frames are transmitted and received through hardware frame queues, which can be grouped in channels for the purpose of hardware scheduling. The Ethernet driver enqueues TX frames on egress queues and after transmission is complete a TX confirmation frame is sent back to the CPU.

When frames are available on ingress queues, a data availability notification is sent to the CPU; notifications are raised per channel, so even if multiple queues in the same channel have available frames, only one notification is sent. After a channel fires a notification, is must be explicitly rearmed.

Each network interface can have multiple Rx, Tx and confirmation queues affined to CPUs, and one channel (DPCON) for each CPU that services at least one queue. DPCONs are used to distribute ingress traffic to different CPUs via the cores' affine DPIOs.

The role of hardware buffer pools is storage of ingress frame data. Each network interface has a privately owned buffer pool which it seeds with kernel allocated buffers.

DPNIs are decoupled from PHYs; a DPNI can be connected to a PHY through a DPMAC object or to another DPNI through an internal link, but the connection is managed by MC and completely transparent to the Ethernet driver.

```
    ---------     ---------     ---------
   | eth if1 |   | eth if2 |   | eth ifn |
    ---------     ---------     ---------
         .            .          .
         .            .          .
         .            .          .
        ---------------------------
       |   DPAA2 Ethernet Driver    |
        ---------------------------
         .            .          .
         .            .          .
         .            .          .
      ------      ------      ------            -------
     | DPNI |    | DPNI |    | DPNI |          | DPMAC |----+
      ------      ------      ------            -------     |
        |            |           |                   |      |
        |            |           |                   |    -----
         ===========              ==================     | PHY |
                                                          -----
```

## Creating a Network Interface

A net device is created for each DPNI object probed on the MC bus. Each DPNI has a number of properties which determine the network interface configuration options and associated hardware resources.

DPNI objects (and the other DPAA2 objects needed for a network interface) can be added to a container on the MC bus in one of two ways: statically, through a Datapath Layout Binary file (DPL) that is parsed by MC at boot time; or created dynamically at runtime, via the DPAA2 objects APIs.

## Features & Offloads

Hardware checksum offloading is supported for TCP and UDP over IPv4/6 frames. The checksum offloads can be independently configured on RX and TX through ethtool.

Hardware offload of unicast and multicast MAC filtering is supported on the ingress path and permanently enabled.

Scatter-gather frames are supported on both RX and TX paths. On TX, SG support is configurable via ethtool; on RX it is always enabled.

The DPAA2 hardware can process jumbo Ethernet frames of up to 10K bytes.

The Ethernet driver defines a static flow hashing scheme that distributes traffic based on a 5-tuple key: src IP, dst IP, IP proto, L4 src port, L4 dst port. No user configuration is supported for now.

Hardware specific statistics for the network interface as well as some non-standard driver stats can be consulted through ethtool -S option.