phylink

Overview

phylink is a mechanism to support hot-pluggable networking modules directly connected to a MAC without needing to re-initialise the adapter on hot-plug events.

phylink supports conventional phylib-based setups, fixed link setups and SFP (Small Formfactor Pluggable) modules at present.

Modes of operation

phylink has several modes of operation, which depend on the firmware settings.

1. PHY mode

   In PHY mode, we use phylib to read the current link settings from the PHY, and pass them to the MAC driver. We expect the MAC driver to configure exactly the modes that are specified without any negotiation being enabled on the link.

2. Fixed mode

   Fixed mode is the same as PHY mode as far as the MAC driver is concerned.

3. In-band mode

   In-band mode is used with 802.3z, SGMII and similar interface modes, and we are expecting to use and honor the in-band negotiation or control word sent across the serdes channel.

By example, what this means is that:

System Message: WARNING/2 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master) (Documentation) (networking)sfp-phylink.rst`, line 43)

Cannot analyze code. No Pygments lexer found for "none".

```
.. code-block:: none

  &eth {
    phy = <&phy>;
    phy-mode = "sgmii";
  };
```

does not use in-band SGMII signalling. The PHY is expected to follow exactly the settings given to it in its :c:func:`mac_config` function. The link should be forced up or down appropriately in the :c:func:`mac_link_up` and :c:func:`mac_link_down` functions.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master) (Documentation) (networking)sfp-phylink.rst`, line 50); *backlink*

Unknown interpreted text role "c:func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master) (Documentation) (networking)sfp-phylink.rst`, line 50); *backlink*

Unknown interpreted text role "c:func".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master) (Documentation) (networking)sfp-phylink.rst`, line 50); *backlink*

Unknown interpreted text role "c:func".

System Message: WARNING/2 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master) (Documentation) (networking)sfp-phylink.rst`, line 55)

Cannot analyze code. No Pygments lexer found for "none".

```
.. code-block:: none

  &eth {
    managed = "in-band-status";
    phy = <&phy>;
    phy-mode = "sgmii";
  };
```

uses in-band mode, where results from the PHY's negotiation are passed to the MAC through the SGMII control word, and the MAC is expected to acknowledge the control word. The :c:func:`mac_link_up` and :c:func:`mac_link_down` functions must not force the MAC side link up and down.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst, line 63); *backlink*

Unknown interpreted text role "c:func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst, line 63); *backlink*

Unknown interpreted text role "c:func".

Rough guide to converting a network driver to sfp/phylink

This guide briefly describes how to convert a network driver from phylib to the sfp/phylink support. Please send patches to improve this documentation.

1. Optionally split the network driver's phylib update function into two parts dealing with link-down and link-up. This can be done as a separate preparation commit.

   An older example of this preparation can be found in git commit fc548b991fb0, although this was splitting into three parts; the link-up part now includes configuring the MAC for the link settings. Please see :c:func:`mac_link_up` for more information on this.

   System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst, line 80); *backlink*

   Unknown interpreted text role "c:func".

2. Replace:

   ```
   select FIXED_PHY
   select PHYLIB
   ```

   with:

   ```
   select PHYLINK
   ```

   in the driver's Kconfig stanza.

3. Add:

   ```
   #include <linux/phylink.h>
   ```

   to the driver's list of header files.

4. Add:

   ```
   struct phylink *phylink;
   struct phylink_config phylink_config;
   ```

   to the driver's private data structure. We shall refer to the driver's private data pointer as `priv` below, and the driver's private data structure as `struct foo_priv`.

5. Replace the following functions:

   System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst, line 113)

Unknown directive type "flat-table".

```
.. flat-table::
 :header-rows: 1
 :widths: 1 1
 :stub-columns: 0

 * - Original function
   - Replacement function
 * - phy_start(phydev)
   - phylink_start(priv->phylink)
 * - phy_stop(phydev)
   - phylink_stop(priv->phylink)
 * - phy_mii_ioctl(phydev, ifr, cmd)
   - phylink_mii_ioctl(priv->phylink, ifr, cmd)
 * - phy_ethtool_get_wol(phydev, wol)
   - phylink_ethtool_get_wol(priv->phylink, wol)
 * - phy_ethtool_set_wol(phydev, wol)
   - phylink_ethtool_set_wol(priv->phylink, wol)
 * - phy_disconnect(phydev)
   - phylink_disconnect_phy(priv->phylink)
```

Please note that some of these functions must be called under the rtnl lock, and will warn if not. This will normally be the case, except if these are called from the driver suspend/resume paths.

6. Add/replace ksettings get/set methods with:

```
static int foo_ethtool_set_link_ksettings(struct net_device *dev,
                                          const struct ethtool_link_ksettings *cmd)
{
        struct foo_priv *priv = netdev_priv(dev);

        return phylink_ethtool_ksettings_set(priv->phylink, cmd);
}

static int foo_ethtool_get_link_ksettings(struct net_device *dev,
                                          struct ethtool_link_ksettings *cmd)
{
        struct foo_priv *priv = netdev_priv(dev);

        return phylink_ethtool_ksettings_get(priv->phylink, cmd);
}
```

7. Replace the call to:

```
phy_dev = of_phy_connect(dev, node, link_func, flags, phy_interface);
```

and associated code with a call to:

```
err = phylink_of_phy_connect(priv->phylink, node, flags);
```

For the most part, `flags` can be zero; these flags are passed to the phy_attach_direct() inside this function call if a PHY is specified in the DT node `node`.

`node` should be the DT node which contains the network phy property, fixed link properties, and will also contain the sfp property.

The setup of fixed links should also be removed; these are handled internally by phylink.

of_phy_connect() was also passed a function pointer for link updates. This function is replaced by a different form of MAC updates described below in (8).

Manipulation of the PHY's supported/advertised happens within phylink based on the validate callback, see below in (8).

Note that the driver no longer needs to store the `phy_interface`, and also note that `phy_interface` becomes a dynamic property, just like the speed, duplex etc. settings.

Finally, note that the MAC driver has no direct access to the PHY anymore; that is because in the phylink model, the PHY can be dynamic.

8. Add a :c:type:`struct phylink_mac_ops <phylink_mac_ops>` instance to the driver, which is a table of function pointers, and implement these functions. The old link update function for :c:func:`of_phy_connect` becomes three methods: :c:func:`mac_link_up`, :c:func:`mac_link_down`, and :c:func:`mac_config`. If step 1 was performed, then the functionality will have been split there.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst`, **line 190**); *backlink*

Unknown interpreted text role "c:type".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst, line 190); *backlink*

Unknown interpreted text role "c:func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst, line 190); *backlink*

Unknown interpreted text role "c:func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst, line 190); *backlink*

Unknown interpreted text role "c:func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst, line 190); *backlink*

Unknown interpreted text role "c:func".

It is important that if in-band negotiation is used, :c:func:`mac_link_up` and :c:func:`mac_link_down` do not prevent the in-band negotiation from completing, since these functions are called when the in-band link state changes - otherwise the link will never come up.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst, line 197); *backlink*

Unknown interpreted text role "c:func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst, line 197); *backlink*

Unknown interpreted text role "c:func".

The :c:func:`validate` method should mask the supplied supported mask, and `state->advertising` with the supported ethtool link modes. These are the new ethtool link modes, so bitmask operations must be used. For an example, see drivers/net/ethernet/marvell/mvneta.c.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst, line 203); *backlink*

Unknown interpreted text role "c:func".

The :c:func:`mac_link_state` method is used to read the link state from the MAC, and report back the settings that the MAC is currently using. This is particularly important for in-band negotiation methods such as 1000base-X and SGMII.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst, line 208); *backlink*

Unknown interpreted text role "c:func".

The :c:func:`mac_link_up` method is used to inform the MAC that the link has come up. The call includes the negotiation mode and interface for reference only. The finalised link parameters are also supplied (speed, duplex and flow control/pause enablement settings) which should be used to configure the MAC when the MAC and PCS are not tightly integrated, or when the settings are not coming from in-band negotiation.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst, line 213); *backlink*

Unknown interpreted text role "c:func".

The :c:func:`mac_config` method is used to update the MAC with the requested state, and must avoid unnecessarily taking the link down when making changes to the MAC configuration. This means the function should modify the state and only take the link down when absolutely necessary to change the MAC configuration. An example of how to do this can be found in :c:func:`mvneta_mac_config` in drivers/net/ethernet/marvell/mvneta.c.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst, line 221); *backlink*

Unknown interpreted text role "c:func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst, line 221); *backlink*

Unknown interpreted text role "c:func".

For further information on these methods, please see the inline documentation in :c:type:`struct phylink_mac_ops <phylink_mac_ops>`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst, line 229); *backlink*

Unknown interpreted text role "c:type".

9. Remove calls to of_parse_phandle() for the PHY, of_phy_register_fixed_link() for fixed links etc. from the probe function, and replace with:

```
struct phylink *phylink;
priv->phylink_config.dev = &dev.dev;
priv->phylink_config.type = PHYLINK_NETDEV;

phylink = phylink_create(&priv->phylink_config, node, phy_mode, &phylink_ops);
if (IS_ERR(phylink)) {
        err = PTR_ERR(phylink);
        fail probe;
}

priv->phylink = phylink;
```

and arrange to destroy the phylink in the probe failure path as appropriate and the removal path too by calling:

```
phylink_destroy(priv->phylink);
```

10. Arrange for MAC link state interrupts to be forwarded into phylink, via:

```
phylink_mac_change(priv->phylink, link_is_up);
```

where `link_is_up` is true if the link is currently up or false otherwise. If a MAC is unable to provide these interrupts, then it should set `priv->phylink_config.pcs_poll = true;` in step 9.

11. Verify that the driver does not call:

```
netif_carrier_on()
netif_carrier_off()
```

as these will interfere with phylink's tracking of the link state, and cause phylink to omit calls via the :c:func:`mac_link_up` and :c:func:`mac_link_down` methods.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst, line 273); *backlink*

Unknown interpreted text role "c:func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst, line 273); *backlink*

Unknown interpreted text role "c:func".

Network drivers should call phylink_stop() and phylink_start() via their suspend/resume paths, which ensures that the appropriate :c:type:`struct phylink_mac_ops <phylink_mac_ops>` methods are called as necessary.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\(linux-master)(Documentation)(networking)sfp-phylink.rst, line 277); *backlink*

Unknown interpreted text role "c:type".

For information describing the SFP cage in DT, please see the binding documentation in the kernel source tree `Documentation/devicetree/bindings/net/sff,sfp.txt`