

Broadcom RoboSwitch Ethernet switch driver

The Broadcom RoboSwitch Ethernet switch family is used in quite a range of xDSL router, cable modems and other multimedia devices.

The actual implementation supports the devices BCM5325E, BCM5365, BCM539x, BCM53115 and BCM53125 as well as BCM63XX.

Implementation details

The driver is located in `drivers/net/dsa/b53/` and is implemented as a DSA driver; see `Documentation/networking/dsa/dsa.rst` for details on the subsystem and what it provides.

The switch is, if possible, configured to enable a Broadcom specific 4-bytes switch tag which gets inserted by the switch for every packet forwarded to the CPU interface, conversely, the CPU network interface should insert a similar tag for packets entering the CPU port. The tag format is described in `net/dsa/tag_brcm.c`.

The configuration of the device depends on whether or not tagging is supported.

The interface names and example network configuration are used according the configuration described in the [ref`dsa-config-showcases`](#).

```
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\dsa\[linux-master] [Documentation] [networking] [dsa]b53.rst, line 29); backlink
```

Unknown interpreted text role "ref".

Configuration with tagging support

The tagging based configuration is desired. It is not specific to the b53 DSA driver and will work like all DSA drivers which supports tagging.

See [ref`dsa-tagged-configuration`](#).

```
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\dsa\[linux-master] [Documentation] [networking] [dsa]b53.rst, line 38); backlink
```

Unknown interpreted text role "ref".

Configuration without tagging support

Older models (5325, 5365) support a different tag format that is not supported yet. 539x and 531x5 require managed mode and some special handling, which is also not yet supported. The tagging support is disabled in these cases and the switch need a different configuration.

The configuration slightly differ from the [ref`dsa-vlan-configuration`](#).

```
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\dsa\[linux-master] [Documentation] [networking] [dsa]b53.rst, line 48); backlink
```

Unknown interpreted text role "ref".

The b53 tags the CPU port in all VLANs, since otherwise any PVID untagged VLAN programming would basically change the CPU port's default PVID and make it untagged, undesirable.

In difference to the configuration described in [ref`dsa-vlan-configuration`](#) the default VLAN 1 has to be removed from the slave interface configuration in single port and gateway configuration, while there is no need to add an extra VLAN configuration in the bridge showcase.

```
System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\networking\dsa\[linux-master] [Documentation] [networking] [dsa]b53.rst, line 54); backlink
```

Unknown interpreted text role "ref".

single port

The configuration can only be set up via VLAN tagging and bridge setup. By default packages are tagged with vid 1:

```
# tag traffic on CPU port
ip link add link eth0 name eth0.1 type vlan id 1
ip link add link eth0 name eth0.2 type vlan id 2
ip link add link eth0 name eth0.3 type vlan id 3

# The master interface needs to be brought up before the slave ports.
ip link set eth0 up
ip link set eth0.1 up
ip link set eth0.2 up
ip link set eth0.3 up

# bring up the slave interfaces
ip link set wan up
ip link set lan1 up
ip link set lan2 up

# create bridge
ip link add name br0 type bridge

# activate VLAN filtering
ip link set dev br0 type bridge vlan_filtering 1

# add ports to bridges
ip link set dev wan master br0
ip link set dev lan1 master br0
ip link set dev lan2 master br0

# tag traffic on ports
bridge vlan add dev lan1 vid 2 pvid untagged
bridge vlan del dev lan1 vid 1
bridge vlan add dev lan2 vid 3 pvid untagged
bridge vlan del dev lan2 vid 1

# configure the VLANs
ip addr add 192.0.2.1/30 dev eth0.1
ip addr add 192.0.2.5/30 dev eth0.2
ip addr add 192.0.2.9/30 dev eth0.3

# bring up the bridge devices
ip link set br0 up
```

bridge

```
# tag traffic on CPU port
ip link add link eth0 name eth0.1 type vlan id 1

# The master interface needs to be brought up before the slave ports.
ip link set eth0 up
ip link set eth0.1 up

# bring up the slave interfaces
ip link set wan up
ip link set lan1 up
ip link set lan2 up

# create bridge
ip link add name br0 type bridge

# activate VLAN filtering
ip link set dev br0 type bridge vlan_filtering 1

# add ports to bridge
ip link set dev wan master br0
ip link set dev lan1 master br0
ip link set dev lan2 master br0
ip link set eth0.1 master br0

# configure the bridge
ip addr add 192.0.2.129/25 dev br0

# bring up the bridge
ip link set dev br0 up
```

gateway

```
# tag traffic on CPU port
```

```
ip link add link eth0 name eth0.1 type vlan id 1
ip link add link eth0 name eth0.2 type vlan id 2

# The master interface needs to be brought up before the slave ports.
ip link set eth0 up
ip link set eth0.1 up
ip link set eth0.2 up

# bring up the slave interfaces
ip link set wan up
ip link set lan1 up
ip link set lan2 up

# create bridge
ip link add name br0 type bridge

# activate VLAN filtering
ip link set dev br0 type bridge vlan_filtering 1

# add ports to bridges
ip link set dev wan master br0
ip link set eth0.1 master br0
ip link set dev lan1 master br0
ip link set dev lan2 master br0

# tag traffic on ports
bridge vlan add dev wan vid 2 pvid untagged
bridge vlan del dev wan vid 1

# configure the VLANs
ip addr add 192.0.2.1/30 dev eth0.2
ip addr add 192.0.2.129/25 dev br0

# bring up the bridge devices
ip link set br0 up
```