

```

*****
63361 Tue Jan 29 13:25:12 2019
new/usr/src/uts/common/io/qede/qede_gld.c
10320 qede_gld.c uses assignment, means equality
*****
_____unchanged_portion_omitted_____

1055 int
1056 qede_multicast(qede_t *qede, boolean_t flag, const uint8_t *ptr_mcaddr)
1057 {
1058     int i, ret = DDI_SUCCESS;
1059     qede_mcast_list_entry_t *ptr_mlist;
1060     qede_mcast_list_entry_t *ptr_entry;
1061     int mc_cnt;
1062     unsigned char *mc_macs, *tmpmc;
1063     size_t size;
1064     boolean_t mcmac_exists = B_FALSE;
1065     enum qede_filter_rx_mode_type mode;

1067     if (!ptr_mcaddr) {
1068         cmn_err(CE_NOTE, "Removing all multicast");
1069     } else {
1070         cmn_err(CE_NOTE,
1071             "qede=%p %s multicast: %02x:%02x:%02x:%02x:%02x:%02x",
1072             qede, (flag) ? "Adding" : "Removing", ptr_mcaddr[0],
1073             ptr_mcaddr[1], ptr_mcaddr[2], ptr_mcaddr[3], ptr_mcaddr[4],
1074             ptr_mcaddr[5]);
1075     }

1078     if (flag && (ptr_mcaddr == NULL)) {
1079         cmn_err(CE_WARN, "ERROR: Multicast address not specified");
1080     }
1081     return EINVAL;

1084     /* exceeds addition of mcaddr above limit */
1085     if (flag && (qede->mc_cnt >= MAX_MC_SOFT_LIMIT)) {
1086         qede_info(qede, "Cannot add more than MAX_MC_SOFT_LIMIT");
1087         return ENOENT;
1088     }

1090     size = MAX_MC_SOFT_LIMIT * ETH_ALLEN;

1092     mc_macs = kmem_zalloc(size, KM_NOSLEEP);
1093     if (!mc_macs) {
1094         cmn_err(CE_WARN, "ERROR: Failed to allocate for mc_macs");
1095         return EINVAL;
1096     }

1098     tmpmc = mc_macs;

1100     /* remove all multicast - as flag not set and mcaddr not specified*/
1101     if (!flag && (ptr_mcaddr == NULL)) {
1102         QEDE_LIST_FOR_EACH_ENTRY(ptr_entry,
1103             &qede->mclist.head, qede_mcast_list_entry_t, mclist_entry)
1104         {
1105             if (ptr_entry != NULL) {
1106                 QEDE_LIST_REMOVE(&ptr_entry->mclist_entry,
1107                     &qede->mclist.head);
1108                 kmem_free(ptr_entry,
1109                     sizeof (qede_mcast_list_entry_t) + ETH_ALLEN);
1110             }
1111         }

1113     ret = qede_set_rx_mac_mcast(qede,

```

```

1114         ECOPE_FILTER_REMOVE, mc_macs, 1);
1115         qede->mc_cnt = 0;
1116         goto exit;
1117     }

1119     QEDE_LIST_FOR_EACH_ENTRY(ptr_entry,
1120         &qede->mclist.head, qede_mcast_list_entry_t, mclist_entry)
1121     {
1122         if ((ptr_entry != NULL) &&
1123             IS_ETH_ADDRESS_EQUAL(ptr_mcaddr, ptr_entry->mac)) {
1124             mcmac_exists = B_TRUE;
1125             break;
1126         }
1127     }
1128     if (flag && mcmac_exists) {
1129         ret = DDI_SUCCESS;
1130         goto exit;
1131     } else if (!flag && !mcmac_exists) {
1132         ret = DDI_SUCCESS;
1133         goto exit;
1134     }

1136     if (flag) {
1137         ptr_entry = kmem_zalloc((sizeof (qede_mcast_list_entry_t) +
1138             ETH_ALLEN), KM_NOSLEEP);
1139         ptr_entry->mac = (uint8_t *)ptr_entry +
1140             sizeof (qede_mcast_list_entry_t);
1141         COPY_ETH_ADDRESS(ptr_mcaddr, ptr_entry->mac);
1142         QEDE_LIST_ADD(&ptr_entry->mclist_entry, &qede->mclist.head);
1143     } else {
1144         QEDE_LIST_REMOVE(&ptr_entry->mclist_entry, &qede->mclist.head);
1145         kmem_free(ptr_entry, sizeof (qede_mcast_list_entry_t) +
1146             ETH_ALLEN);
1147     }

1149     mc_cnt = 0;
1150     QEDE_LIST_FOR_EACH_ENTRY(ptr_entry, &qede->mclist.head,
1151         qede_mcast_list_entry_t, mclist_entry) {
1152         COPY_ETH_ADDRESS(ptr_entry->mac, tmpmc);
1153         tmpmc += ETH_ALLEN;
1154         mc_cnt++;
1155     }
1156     qede->mc_cnt = mc_cnt;
1157     if (mc_cnt <= 64) {
1158         ret = qede_set_rx_mac_mcast(qede, ECOPE_FILTER_ADD,
1159             (unsigned char *)mc_macs, mc_cnt);
1160         if ((qede->params.multi_promisc_fl == B_TRUE) &&
1161             (qede->params.promisc_fl == B_FALSE)) {
1162             mode = QEDE_FILTER_RX_MODE_REGULAR;
1163             ret = qede_set_filter_rx_mode(qede, mode);
1164         }
1165         qede->params.multi_promisc_fl = B_FALSE;
1166     } else {
1167         if ((qede->params.multi_promisc_fl == B_FALSE) &&
1168             (qede->params.promisc_fl == B_FALSE)) {
1169             ret = qede_set_filter_rx_mode(qede,
1170                 QEDE_FILTER_RX_MODE_MULTI_PROMISC);
1171         }
1172         qede->params.multi_promisc_fl = B_TRUE;
1173         qede_info(qede, "mode is MULTI_PROMISC");
1174     }

1175     exit:
1176     kmem_free(mc_macs, size);
1177     qede_info(qede, "multicast ret %d mc_cnt %d\n", ret, qede->mc_cnt);
1178     return (ret);

```

new/usr/src/uts/common/io/qede/qede_gld.c

3

1179 }

unchanged_portion_omitted

new/usr/src/uts/common/io/qede/qede_version.h

1

1403 Tue Jan 29 13:25:12 2019

new/usr/src/uts/common/io/qede/qede_version.h

10320 qede_gld.c uses assignment, means equality

```
1 /*
2  * CDDL HEADER START
3  *
4  * The contents of this file are subject to the terms of the
5  * Common Development and Distribution License, v.1, (the "License").
6  * You may not use this file except in compliance with the License.
7  *
8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9  * or http://opensource.org/licenses/CDDL-1.0.
10 * See the License for the specific language governing permissions
11 * and limitations under the License.
12 *
13 * When distributing Covered Code, include this CDDL HEADER in each
14 * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 * If applicable, add the following below this CDDL HEADER, with the
16 * fields enclosed by brackets "[]" replaced with your own identifying
17 * information: Portions Copyright [yyyy] [name of copyright owner]
18 *
19 * CDDL HEADER END
20 */

22 /*
23 * Copyright 2014-2017 Cavium, Inc.
24 * The contents of this file are subject to the terms of the Common Development
25 * and Distribution License, v.1, (the "License").

27 * You may not use this file except in compliance with the License.

29 * You can obtain a copy of the License at available
30 * at http://opensource.org/licenses/CDDL-1.0

32 * See the License for the specific language governing permissions and
33 * limitations under the License.
34 */

37 #ifndef _QEDE_VERSION_H
38 #define _QEDE_VERSION_H

40 #define MAJVERSION 8
41 #define MINVERSION 1
42 #define REVVERSION 24
42 #define REVVERSION 23

44 #endif /* !_QEDE_VERSION_H */
```