Subject: [PATCH v7 06/10] qlge driver: rename internal SEM_SET macro to SEM_INIT
Posted by Stanislav Kinsbursky on Thu, 18 Oct 2012 10:23:06 GMT
View Forum Message <> Reply to Message

The reason for this patch is that SET_SET is desired to be a new part IPC
sys_semctl() API.
The name itself for IPC is quite natural, because all linux-specific commands
names for IPC system calls are originally created by replacing "IPC_" part by
"SEM_"("MSG_", "SHM_") part.
So, I'm hoping, that this change doesn't really matters for "QLogic qlge NIC
HBA Driver" developers, since it's just an internal define.

Signed-off-by: Stanislav Kinsbursky <skinsbursky@parallels.com>
---
 drivers/net/ethernet/qlogic/qlge/qlge.h | 4 ++--
 drivers/net/ethernet/qlogic/qlge/qlge_main.c | 16 ++++++++--------
 2 files changed, 10 insertions(+), 10 deletions(-)

diff --git a/drivers/net/ethernet/qlogic/qlge/qlge.h b/drivers/net/ethernet/qlogic/qlge/qlge.h
index a131d7b..6f46ea5 100644
--- a/drivers/net/ethernet/qlogic/qlge/qlge.h
+++ b/drivers/net/ethernet/qlogic/qlge/qlge.h
@@ -347,10 +347,10 @@ enum {
 enum {
  /*
   * Example:
- * reg = SEM_XGMAC0_MASK | (SEM_SET << SEM_XGMAC0_SHIFT)
+ * reg = SEM_XGMAC0_MASK | (SEM_INIT << SEM_XGMAC0_SHIFT)
   */
  SEM_CLEAR = 0,
- SEM_SET = 1,
+ SEM_INIT = 1,
  SEM_FORCE = 3,
  SEM_XGMAC0_SHIFT = 0,
  SEM_XGMAC1_SHIFT = 2,
diff --git a/drivers/net/ethernet/qlogic/qlge/qlge_main.c
b/drivers/net/ethernet/qlogic/qlge/qlge_main.c
index b262d61..cfb0f62 100644
--- a/drivers/net/ethernet/qlogic/qlge/qlge_main.c
+++ b/drivers/net/ethernet/qlogic/qlge/qlge_main.c
@@ -109,28 +109,28 @@ static int ql_sem_trylock(struct ql_adapter *qdev, u32 sem_mask)

  switch (sem_mask) {
  case SEM_XGMAC0_MASK:
- sem_bits = SEM_SET << SEM_XGMAC0_SHIFT;
+ sem_bits = SEM_INIT << SEM_XGMAC0_SHIFT;
   break;

```
  case SEM_XGMAC1_MASK:
-  sem_bits = SEM_SET << SEM_XGMAC1_SHIFT;
+  sem_bits = SEM_INIT << SEM_XGMAC1_SHIFT;
   break;
  case SEM_ICB_MASK:
-  sem_bits = SEM_SET << SEM_ICB_SHIFT;
+  sem_bits = SEM_INIT << SEM_ICB_SHIFT;
   break;
  case SEM_MAC_ADDR_MASK:
-  sem_bits = SEM_SET << SEM_MAC_ADDR_SHIFT;
+  sem_bits = SEM_INIT << SEM_MAC_ADDR_SHIFT;
   break;
  case SEM_FLASH_MASK:
-  sem_bits = SEM_SET << SEM_FLASH_SHIFT;
+  sem_bits = SEM_INIT << SEM_FLASH_SHIFT;
   break;
  case SEM_PROBE_MASK:
-  sem_bits = SEM_SET << SEM_PROBE_SHIFT;
+  sem_bits = SEM_INIT << SEM_PROBE_SHIFT;
   break;
  case SEM_RT_IDX_MASK:
-  sem_bits = SEM_SET << SEM_RT_IDX_SHIFT;
+  sem_bits = SEM_INIT << SEM_RT_IDX_SHIFT;
   break;
  case SEM_PROC_REG_MASK:
-  sem_bits = SEM_SET << SEM_PROC_REG_SHIFT;
+  sem_bits = SEM_INIT << SEM_PROC_REG_SHIFT;
   break;
  default:
   netif_alert(qdev, probe, qdev->ndev, "bad Semaphore mask!.\n");
```