

開発支援環境における NIC ドライバの改変箇所まとめ

2016/5/12

藤田将輝

1 はじめに

開発支援環境を実装するにあたり，NIC ドライバを改変した．本資料では，開発支援環境の実装における NIC ドライバの改変目的と改変箇所についてまとめている．この改変については，GitHub の [fujita-m/Mint](#) の [a392851ade397bdadc216c493dea48687f5b2cc8](#)(コミット ID) に示されている．

2 実装環境

実装環境を表 1 に示す．

表 1 実装環境

項目名	環境
OS	Fedora14 x86_64(Mint 3.0.8)
CPU	Intel(R) Core(TM) Core i7-870 @ 2.93GHz
メモリ	PC3-10600 DDR3-SDRAM (dual channel) @ 165.6Gbps
Chipset	Intel P55 @ 16Gbps
NIC ドライバ	RTL8169
ソースコード	r8169.c

3 改変目的

本開発支援環境では，NIC を用いずに環境を構築する．このため，以下の改変が必要になる．

(1) 割込契機の変更

本環境では IPI を用いて，割込を発生させる．このため，IPI によって指定したベクタ番号で NIC ドライバの割込ハンドラが動作するように改変する必要がある．

(2) 受信ディスクリプタ，受信バッファのアドレス変更

本環境では NIC ドライバのパケット受信割込処理を再現する．このため，開発支援 OS と開発対象 OS でメモリを共有し，ここに受信ディスクリプタと受信バッファを作成することでパケットを授受する必要がある．

(3) 割込ステータスの調整

NIC ドライバの割込ハンドラが動作すると NIC からの割込を禁止し、パケットの受信処理が終了すると割込を許可する。この割込の禁止/許可のステータスは NIC ハードウェアのレジスタによって表されている。したがって、このステータスを開発支援 OS と開発対象 OS で共有する必要がある。

(4) NIC ハードウェアの停止

本環境では NIC ハードウェアは使用しないという方針がある。このため、NIC ドライバの初期化処理の最後で NIC を停止させる必要がある。

4 改変箇所

4.1 割込契機の変更

割込契機を IPI とし、指定したベクタ番号で NIC ドライバの割込ハンドラを動作させるよう改変を加える。NIC ドライバは、初期化処理である `rtl8169_open()` で割込ハンドラを登録する。このため、登録した割込ハンドラの IRQ 番号をベクタ管理表の指定したエントリに登録する処理を追加する。追加する処理を以下に示す。

(追加) ベクタ管理票の指定したエントリに登録した IRQ 番号を格納する。

4.2 受信ディスクリプタ、受信バッファのアドレス変更

Mint の共有メモリに NIC ドライバの受信ディスクリプタと受信バッファを作成するよう改変を加える。受信ディスクリプタと受信バッファそれぞれの改変箇所を以下に示す。

(1) 受信ディスクリプタ

`rtl8169_open()` 中で初期化される。受信ディスクリプタの先頭アドレスを決定する際に Mint の共有メモリのアドレスとなるよう改変を加える。改変前と改変後を以下に示す。

(改変前) 受信ディスクリプタの先頭アドレスを DMA 用のアドレスから選定し、決定する。

(改変後) 受信ディスクリプタの先頭アドレスを Mint の共有メモリのアドレスとする。

(2) 受信バッファ

受信バッファのアドレスは受信ディスクリプタの初期化時に決定され、受信ディスクリプタに格納される。アドレスの決定は、`rtl8169_alloc_rx_data()` で行われる。このため、この関数を改変し、受信バッファのアドレスを Mint の共有メモリのアドレスとした。改変前と改変後を以下に示す。

(改変前) 受信バッファのアドレスを DMA 用のアドレスから選定し、決定する。

(改変後) 受信バッファのアドレスを Mint の共有メモリのアドレスとする。

4.3 割込ステータスの調整

NIC ドライバの割込ハンドラが動作した際，カーネルにパケットの処理を依頼し，割込を禁止する．パケットの処理が終わると割込を許可する．これを再現するため，NIC の割込を禁止するタイミングで開発支援 OS からの割込を禁止する必要がある．通常，NIC のレジスタの値で割込の禁止/許可を示すが，これでは開発支援 OS からの割込を禁止できない．このため，Mint の共有メモリに割込の禁止/許可を示すフラグを用意し，フラグが立っている場合は割込を禁止し，フラグが立っていなければ割込を許可することとした．このフラグの更新を行うのは NIC ドライバが割込を禁止/許可するタイミングと同じタイミングである．割込の禁止を行うのは `rtl8169_interrupt()` の処理中であり，許可を行うのは `rtl8169_poll()` の処理中である．追加した処理を以下に示す．

(追加) 割込の禁止/許可のタイミングでフラグを更新する．

4.4 NIC ハードウェアの停止

本環境の方針で NIC ハードウェアは使用しないため，ネットワークインタフェースを起動した直後に NIC を停止するように改変する必要がある．NIC ドライバには NIC ハードウェアの動作を停止させる関数 `rtl8169_asic_down()` がある．この関数を `rtl8169_open()` の処理の最後に加えることにより，NIC ハードウェアの停止を行う．追加した処理を以下に示す．

(追加) 初期化処理の最後で `rtl8169_asic_down()` を動作させる．

5 関連関数

4 章で示した 5 つの関数について以下に示す．

(1) `rtl8169_open()`

【形式】

```
static int rtl8169_open(struct net_device *dev)
```

【引数】

`struct net_device *dev` : ネットデバイス構造体

【戻り値】

成功 : 0

失敗 : 0 以外

【機能】

パケットの送受信ディスクリプタやバッファの初期化等を行う．また，割込ハンドラの登録を行う．その他，ハードウェアの初期化を行う．

(2) `rtl8169_alloc_rx_data()`

【形式】

```
static struct sk_buff *rtl8169_alloc_rx_data(struct rtl8169_private *tp, struct RxDesc *desc)
```

【引数】

struct rtl8169_private *tp : NIC ドライバのプライベート構造体
struct RxDesc *desc : 受信ディスクリプタ

【戻り値】

成功 : アロケートした領域へのポインタ
失敗 : NULL

【機能】

受信バッファのアドレスを決定し、受信ディスクリプタに受信バッファのアドレスを登録する。また、受信バッファの領域をアロケートする。

(3) rtl8169_interrupt()**【形式】**

```
static irqreturn_t rtl8169_interrupt(int irq, void *dev_instance)
```

【引数】

int irq : IRQ 番号 void *dev_instance : ネットデバイス構造体

【戻り値】

IRQ_RETVAL() の戻り値

【機能】

パケットの処理をカーネルに依頼する。この際、NIC からの割込を禁止する。

(4) rtl8169_poll()**【形式】**

```
static int rtl8169_poll(struct napi_struct *napi, int budget)
```

【引数】

struct napi_struct *napi : NewAPI 構造体
int budget : 一度に処理できる最大のパケット数

【戻り値】

処理したパケット数

【機能】

パケット受信処理を呼び出し、パケットをソケットバッファに格納した後、上位層に上げる。全てのパケットの処理が終われば割込を許可する。

(5) rtl8169_asic_down()**【形式】**

```
static void rtl8169_asic_down(void __iomem *ioaddr)
```

【引数】

void __iomem *ioaddr : 入出力アドレス

【戻り値】

なし

【機能】

NIC ハードウェアの各種レジスタをクリアし，停止させる．

6 おわりに

本資料では，本開発支援環境における NIC ドライバの改変目的と改変箇所についてまとめた．今後は，GitHub における NIC ドライバの過去のコミットから修正されたバグを調査し，本開発支援環境で再現できるバグとそうでないバグを仕分ける．これにより，開発支援環境に追加すべき機能をまとめる．