

開発支援環境における NIC ドライバの改変箇所まとめ (修正版)

2016/5/31

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 `dev_support` ブランチにおけるコミット `3931b69c66d7447a4f2f7cff18175145eb1a0164`(コミット ID) に示されている．前回の資料では以下の内容が記載されておらず，目的や改変量が不明確であった．このため，これを追記した．

- (1) それぞれの改変が全体として何を実現するのか．
- (2) 関連関数と改変箇所の対応．
- (3) 既存関数のどの部分を改変し，どの程度改変を加えたか．

2 修正箇所

前回の資料からの修正箇所について以下に示す．

- (1) それぞれの改変が全体として何を実現するかを追記
(修正前) 4 章改変目的において，その改変が開発支援環境全体でどのような機能を実現するかを記載していない．
(修正後) 4 章改変目的において，その改変が全体としてどのような機能を実現するものを記載した．
- (2) 改変箇所と関連関数を対応
(修正前) 改変箇所と関連関数を別の章に記載しており，対応が不明であった．
(修正後) 改変箇所に関連関数を追記することにより，どの関数を改変したかを明確にした．
- (3) 関数のどの部分を改変し，どの程度改変を加えたかを追記
(修正前) 関連関数のどの部分を改変したかとどの程度改変を加えたかを記載していない．
(修正後) それぞれの改変について，どの関数を何行改変したかを 6 章において表として示した．

3 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

4 改変目的

本開発支援環境では，開発支援 OS が NIC ハードウェアの動作を擬似することで，割込間隔や割込回数を制御可能にする．割込間隔や割込回数はハードウェアに依存するため，制御が困難である．これを開発支援 OS が擬似することにより，ソフトウェアによる制御が可能になり，割込間隔等の指定が容易になる．開発支援 OS の NIC ハードウェアの擬似に NIC ドライバを対応させるためには，NIC ドライバに以下の 4 つの改変が必要になる．

(1) 割込契機の変更

NIC ハードウェアはパケットをメモリに配置し終わると CPU に割込を発生させる．この割込は I/O APIC を通じてコアに通知される．本開発支援環境では NIC ハードウェアを開発支援 OS が擬似するため，この割込を再現する必要がある．本開発支援環境では割込処理の契機をコア間の割込である IPI とした．これにより，NIC ハードウェアを用いずに割込の通知が可能になる．割込対象 OS では IPI によって割込処理が開始するよう NIC ドライバを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

NIC ハードウェアは受信したパケットをメモリに配置する．メモリにパケットを配置する機能は開発支援 OS で実装した．この際，パケットの配置先は NIC ドライバが管理する受信バッファである．また，NIC ドライバは受信バッファの状態を管理する受信ディスクリプタを参照することでパケットの受信処理を行う．NIC ハードウェアはパケットを配置すると受信ディスクリプタを更新し，受信状態にする．これにより，NIC ドライバが受信バッファのどのエントリにパケットを受信したかを認識する．これらのため，受信ディスクリプタと受信バッファを開発支援 OS と開発対象 OS で参照可能にする必要がある．したがって，受信ディスクリプタと受信バッファを Mint の共有メモリに配置することによりこれを実現する．受信ディスクリプタと受信バッファの初期化は NIC ドライバで行うため，初期化時にこれらの配置位置を Mint の共有メモリ内に固定することで両 OS から参照可能にする．

(3) 割込ステータスの調整

NIC ドライバは割込を受けると直ぐにパケット受信処理を行うわけではない．NIC ドライバが割込を受けた際に行うのは，NIC ハードウェアからの割込の禁止とカーネルへのパケット処理の依頼の 2 つのみである．これは割込処理は最小限に止め，他の処理に影響しないようにするという指向によるものである．NIC ハードウェアからの割込の禁止は，NIC ハードウェアの割込に関するレジスタが制御している．しかし，本開発支援環境では NIC ハードウェアからの割込ではなく，開発支援 OS からの割込 (IPI) を禁止しなければならない．このため，割込の禁止/許可を表すフラグを共有メモリに配置し，開発支援 OS と開発対象 OS で参照可能にすることでこれを実現する．フラグが更新されるのは 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．開発支援 OS は IPI を送信する直前にこのフラグを確認し，フラグが立っていれば IPI を送信せず，立っていなければ IPI を送信する．

(4) NIC ハードウェアの停止

本開発支援環境では基本的に NIC ハードウェアを使用しないがデバイスの認識時に限って使用している．認識を終えると NIC ハードウェアは動作する必要が無いため，NIC ドライバの初期化処理の終了時に動作を停止させる．

5 改変箇所

5.1 改変するドライバ

改変を加えるドライバは RTL8169 である．ソースコードは `drivers/net/r8169.c` である．

5.2 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，割込ハンドラを登録した直後に，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) ベクタ管理表の指定したエントリに登録した IRQ 番号を格納する．

`rtl8169_open()` 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。

5.3 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。

rtl8169_alloc_rx_data() について、以下に示す。

(A) 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。

5.4 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。

`rtl8169_interrupt()` と `rtl8169_poll()` について、以下に示す。

(1) `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。

(2) `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。

5.5 NIC ハードウェアの停止

本開発支援環境では基本的に NIC ハードウェアは使用しないが、ネットワークインタフェースの起動時にのみ使用している。ネットワークインタフェースを起動しなければ NIC ドライバを使用できないためである。ネットワークインタフェースを起動した後は NIC ハードウェアを使用する必要はないため、NIC ハードウェアの動作を停止させる。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。

`rtl8169_asic_down()` について、以下に示す。

(1) `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 改変量

それぞれの改変における改変を加えた関数と改変量を表 2 に示す。

表 2 改変内容と改変量について

改変内容	改変を加えた関数	追加行数	削除行数
割込契機の変更	<code>rtl8169_open()</code>	1	0
受信ディスクリプタ， 受信バッファのアドレス変更	<code>rtl8169_open()</code>	1	2
	<code>rtl8169_alloc_rx_data()</code>	1	2
割込ステータスの調整	<code>rtl8169_interrupt()</code>	2	0
	<code>rtl8169_poll()</code>	2	0
NIC ハードウェアの停止	<code>rtl8169_open()</code>	1	0

7 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。