

# Mint を用いたデバッグ支援環境の評価

2015/10/2

藤田将輝

## 1 はじめに

本資料では，Mint を用いたデバッグ支援環境の評価について述べる．本デバッグ支援環境では IPI を用いて割り込みを発生させている．したがって，まずコアがどの程度の短い間隔であれば IPI を受信できるのかを測定した．次に，本デバッグ支援環境を用いた際，NIC ドライバがどの程度の割り込み間隔であればパケットを正常に処理できるかを測定した．最後に，本デバッグ支援環境を用いた際，デバッグ対象 OS 上で動作する UDP の受信プログラムがどの程度の割り込み間隔であれば，パケットを正常に受信できるかを測定した．これらの測定結果から本デバッグ支援環境を用いた際のスループットを算出し，実際の NIC と比較して，本デバッグ支援環境が NIC ドライバをデバッグすることに有益かどうかを評価する．

## 2 評価目的

本評価の目的は，Mint を用いたデバッグ支援環境を使用して，デバッグ対象 OS にパケット受信割り込み処理をさせた際に，どの程度の割り込み間隔とパケットのサイズならば正常にパケットを処理できるかを測定し，実現できるスループットを算出することにより，実際の NIC を再現することに十分な性能を実現できているかを評価することである．

## 3 評価環境

本評価における評価環境を表 1 に示す．

表 1 測定環境

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	2GB
Chipset	Intel(R) 5 Series/3400
NIC ドライバ	RTL8169
ソースコード	r8169.c

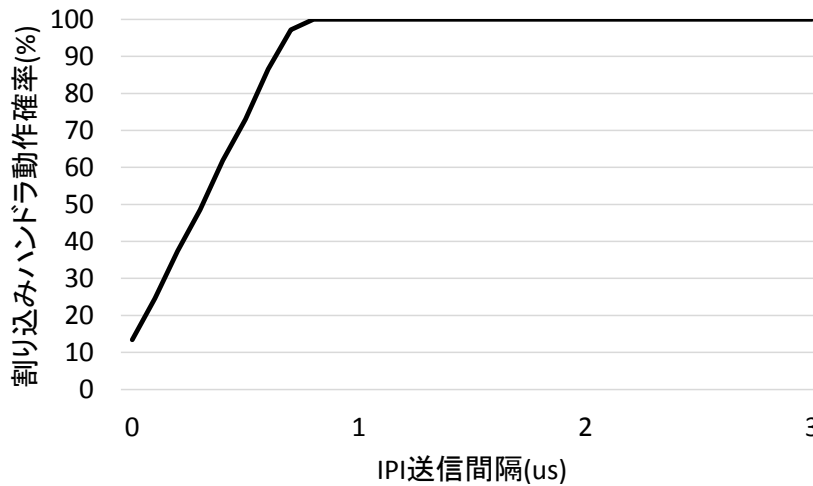


図 1 IPI の反応限界

## 4 測定対象

本評価で行う測定は、以下の 3 つである。

- 【測定 1】 コアが反応可能な IPI の受信間隔の測定
- 【測定 2】 NIC ドライバがパケットを処理可能なデバッグ支援機構の動作間隔とパケットサイズの測定
- 【測定 3】 デバッグ対象 OS 上で動作する UDP の受信プログラムがパケットを処理可能なデバッグ支援機構の動作間隔とパケットサイズの測定

## 5 測定 1

### 5.1 測定方法

IPI の間隔を調整し、連続で IPI を送信し、割り込みハンドラの動作確率を測定する。具体的には、デバッグ支援 OS からデバッグ対象 OS に 5000 回連続で IPI を送信し、デバッグ対象 OS の割り込みハンドラを動作させた際に、カウンタをインクリメントすることで、動作した割り込みハンドラの回数を測定し、動作確率を算出する。これらの動作を 1 サイクルとし、IPI の送信間隔を一定時間増加させながら何サイクルも行う。これにより、全ての IPI にコアが反応できる送信間隔を調査する。

### 5.2 測定結果

測定結果を図 1 に示し、分かったことを以下で説明する。

- (1) IPI の送信間隔が 0us から 1us の間、割り込みハンドラの動作確率は一次関数的に増加する。

- (2) IPI の送信間隔を 1 $\mu$ s 以上にすると動作確率は 100% となる .

## 6 測定 2

### 6.1 測定方法

本デバッグ支援環境を用いて、連続でパケットを送信した際の NIC ドライバでパケットを処理できる確率を測定する。測定に用いたパケットは UDP パケットの Ethernet フレームであり、パケットのサイズは Ethernet フレームのサイズである。具体的には以下の方法で NIC ドライバでのパケットの受信成功確率を測定する。

- (1) デバッグ支援機構の動作間隔を指定し、デバッグ支援機構を 5000 回動作させる。
- (2) NIC ドライバでパケットを処理する際、共有メモリに配置したカウンタをインクリメントする。
- (3) デバッグ支援機構を 5000 回動作させた後、デバッグ支援 OS が共有メモリに配置したカウンタを取得し、成功回数を求める。この回数とデバッグ支援機構動作回数である 5000 回から受信成功率を算出する。

この流れを 1 サイクルとし、1 サイクル毎にデバッグ支援機構の動作間隔を 100ns 増加させながら何サイクルも行ふ。これにより、各サイクルのデバッグ支援機構の動作間隔でのパケット受信成功確率を求められる。また、これらの操作を以下の 6 つのサイズのパケットで行うことで、各パケットのサイズにおける受信可能な動作間隔がわかる。

- (1) 1KB
- (2) 1500B
- (3) 4KB
- (4) 8KB
- (5) 12KB
- (6) 16KB

さらに、1 サイクルにかかった時間を測定する。この時間と、処理したデータ量からどの程度の通信量を実現できているかが分かる。

### 6.2 測定結果

測定結果を図 2 に示し、読み取れることを以下に示す。

- (1) 0 $\mu$ s から 3 $\mu$ s の間、どのサイズでもほとんどの割り込みでパケット受信に失敗している。
- (2) 3 $\mu$ s を超えると、一次関数的に受信成功確率は増加し始める。この際、パケットサイズが小さいほど傾きが急になっている。
- (3) 受信成功確率が 100% になってからはどれだけ動作間隔を増加させても受信成功確率は 100% と

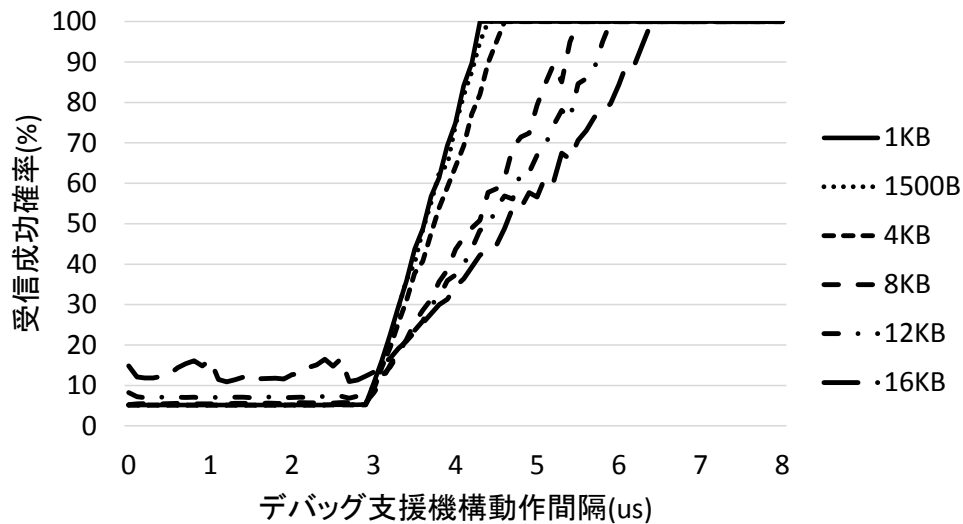


図2 NICドライバにおける受信成功確率

なる。

## 7 測定3

### 7.1 測定方法

本デバッグ支援環境を用いて、デバッグ支援 OS からデバッグ対象 OS へパケットを送信した際の、デバッグ対象 OS 上で動作する UDP の受信用プログラムでどの程度の間隔とパケットサイズならば、パケットを受信できるかを測定する。また、デバッグ支援 OS からデバッグ対象 OS 上のプログラムまでで実現できている通信量を測定する。具体的な方法は、6.1 節とほぼ同様である。カウンタをインクリメントする箇所が 6.1 節では NIC ドライバがパケットを受信した時であったのに対し、本測定では UDP の受信プログラムがパケットを受信した時であることが差分である。

### 7.2 測定結果

測定結果を図3に示し、読み取れることを以下に示す。

- (1) どのパケットサイズでもある時点まではほとんど受信に失敗している。
- (2) ある時点から一次関数的に受信成功確率が増加している。
- (3) 受信成功確率が増加し始めるある時点は、パケットサイズが増加するに連れて増大している。
- (4) 受信成功確率が一次関数的に増加する際、パケットのサイズに関わらず、傾きは一定である。
- (5) 受信成功確率が0回になるデバッグ支援機構の動作間隔以上に間隔を増加させても受信成功確率は100%となる。

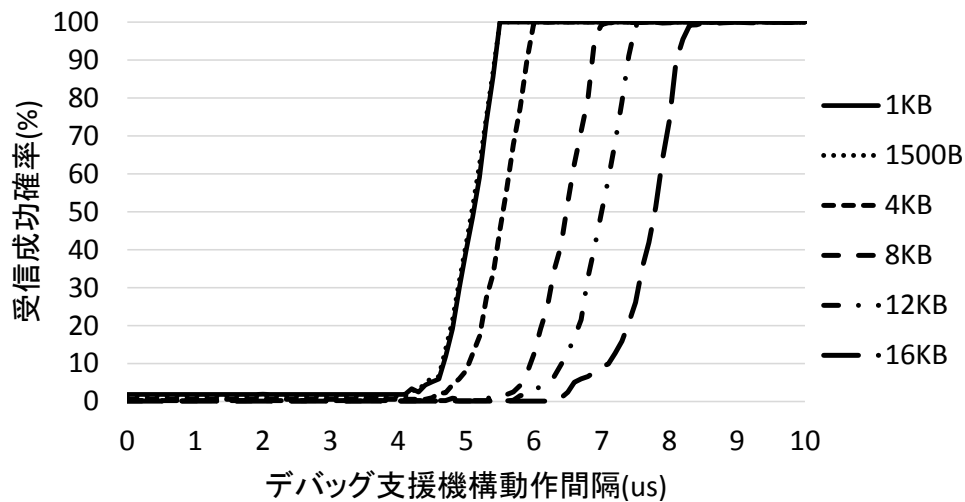


図3 UDP 受信プログラムにおける受信成功確率

## 8 考察

### 8.1 測定 1

測定結果である 5.2 節から，現在の測定環境では IPI の送信間隔が約 1us 以上であれば連続で IPI を送信した際に，全ての IPI に割り込みハンドラが動作できることが分かる．したがって，本実験環境下では，連続で割り込みを発生させる際，最低でも 1us 以上の間隔は必要であると言える．

### 8.2 測定 2

測定結果である 6.2 節から，連続で割り込みを発生させた際，パケットのサイズを増加させるほど全てのパケットを処理することにかかる時間が増加することが分かる．また，NIC ドライバで測定した受信成功回数，パケットのサイズ，および 1 サイクルにかかった時間から各パケットにおける通信量を算出した．各パケットサイズにおける初めて受信成功確率が 100% になった際のデバッグ支援機構の動作間隔，1 サイクルにかかった時間，総データ量，および通信量を表 2 に示す．

### 8.3 測定 3

測定結果である 7.2 節から，連続で割り込みを発生させた際，NIC ドライバでの結果と比べて，より多くの時間がかかっていることが分かる．このことから，NIC ドライバですべてのパケットを処理できていても，プログラムまでパケットが届くにはより多くの時間がかかることが分かる．また，UDP の受信プログラムで測定した受信成功回数，パケットのサイズ，および 1 サイクルにかかった時間から各パケットサイズにおける通信量を算出した．各パケットサイズにおける初めて受信成功確率が 100% になった際のデバッグ支援機構の動作間隔，1 サイクルにかかった時間，総データ量，および通信量を表 3 に示す．

表 2 NIC ドライバにおけるパケットのサイズと割り込み間隔の関係

サイズ	割り込み間隔	1 サイクルの時間	総データ量	スループット
1KB	4.3us	21689us	5000KB	1.7Gbps
1500B	4.4us	22203us	7324KB	2.5Gbps
4KB	4.6us	23195us	20000KB	6.5Gbps
8KB	5.5us	27788us	40000KB	10.9Gbps
12KB	5.9us	29770us	60000KB	15.3Gbps
16KB	6.4us	32491us	80000KB	18.7Gbps

表 3 UDP 受信プログラムにおけるパケットのサイズと割り込み間隔の関係

サイズ	割り込み間隔	1 サイクルの時間	総データ量	スループット
1KB	5.5us	27677us	5000KB	1.3Gbps
1500B	5.5us	27700us	7324KB	2.0Gbps
4KB	6.1us	30705us	20000KB	4.9Gbps
8KB	7.7us	38649us	40000KB	7.8Gbps
12KB	9.4us	47241us	60000KB	9.6Gbps
16KB	11.1us	55789us	80000KB	10.9Gbps

## 9 評価

上記の結果と考察から，NIC ドライバで最大約 18.7Gbps，デバッグ対象 OS 上で動作する UDP 受信プログラムで最大約 10.9Gbps のスループットを実現できることが分かった．PCI Express 1.1 の 1 レーンの転送量が 2.5Gbps であることから，実際の NIC が実現できる最大のスループットは 2.5Gbps であると言える．本デバッグ支援環境で実現できるスループットと実際の NIC が実現できるスループットを比較すると，本デバッグ支援環境は実際の NIC のスループットを大きく超えたスループットを実現できていることがわかる．このことから，本デバッグ支援環境を用いることで現在は実現できていない高スループットの NIC のエミュレートができると考えられる．高スループットの NIC やバスが開発されたと想定し，これらに対応するドライバを開発できる．

## 10 おわりに

本資料では，本デバッグ支援環境の評価について述べた．本デバッグ支援環境を用いると，NIC ドライバで最大 18.7Gbps，プログラムで最大 10.9Gbps を実現できる．実際の NIC は最大 2.5Gbps のスループットであるため，これの 4～7 倍のスループットを実現できている．