

# デバッグ支援環境における受信処理可能な通信量の測定方法

2015/8/25

藤田将輝

## 1 はじめに

本資料では本デバッグ支援環境において、パケットを連続で送信する際、パケットがどの程度の間隔とサイズならば正常にパケット受信処理をできるかの測定方法について記述する。

## 2 測定方法

### 2.1 概要

パケットを 3000 個作成し、3000 回の割り込み処理をさせる操作を 1 サイクルとする。これを 2000 サイクル繰り返し、各サイクルで失敗した回数を求める。サイクル数が増える毎に割り込みの発行間隔を増加させる。また、これらの処理をパケットのサイズを変化させて測定する。さらに、1 サイクル毎の時間も測定することで通信量を求める。1 サイクルの処理を図 1 に示し、以下で説明する。

- (1) デバッグ対象 OS 上で、1 サイクルで最後に受信したパケットのシーケンス番号を表示するプログラムを動作させる。
- (2) デバッグ支援 OS 上で動作するパケットジェネレータにより、3000 個のパケットを作成する。この際、送信する順にパケットにシーケンス番号を付与する。
- (3) システムコールとして実装しているデバッグ支援機構を呼び出す。
- (4) デバッグ支援機構の処理中で以下の処理を 3000 回繰り返す。
  - (A) パケットを共有メモリに配置し、割り込みを発生させる。
  - (B) デバッグ対象 OS の NIC ドライバの割り込みハンドラが動作し、パケット受信割り込み処理を行う。
  - (C) 割り込みハンドラの処理の最後で Mint の共有メモリに配置しているカウンタをインクリメントする。
- (5) 処理を終えると割り込みの回数の 3000 回とカウンタの値を比較することで処理の失敗回数を求める。

### 2.2 測定手順

測定は Mint で 2 つの OS を用いて行う。一方は NIC の擬似をし、割り込みを発生させるデバッグ支援 OS である。他方は割り込みが発生するデバッグ対象 OS である。また、同じ測定方法、手順でパ

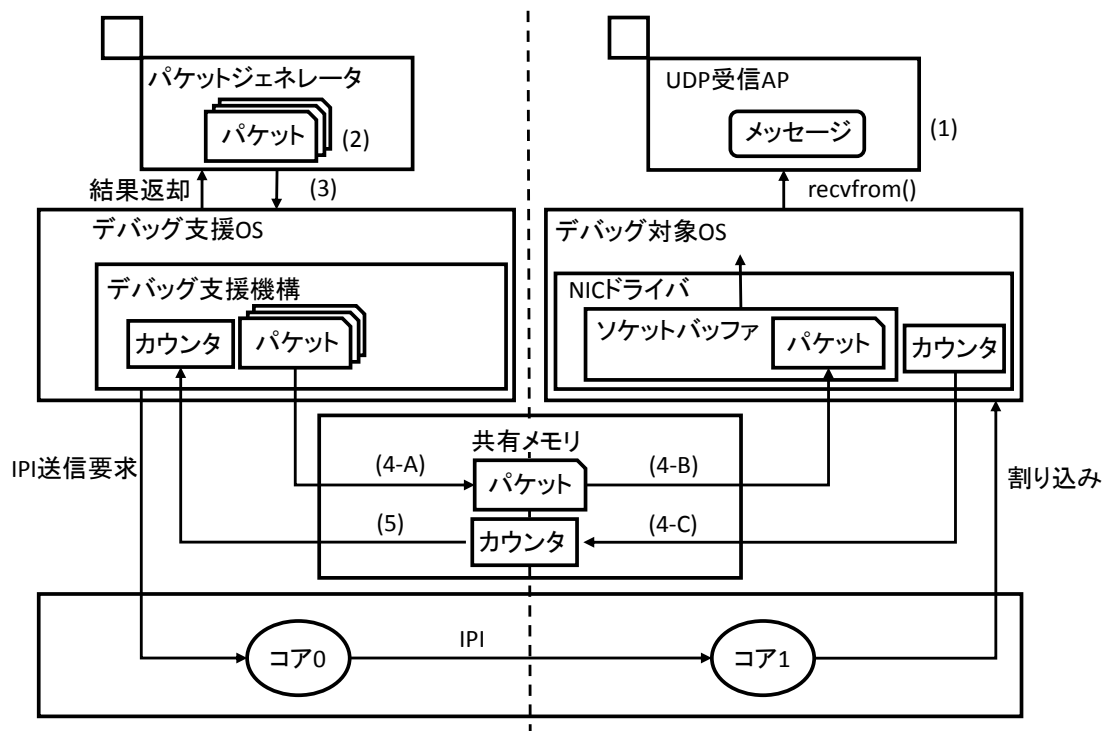


図 1 1 サイクルの処理

ケットのサイズを変化させて、測定を行う。測定の手順を以下に示す。

- (1) Mint を用いてデバッグ支援 OS とデバッグ対象 OS を動作させる。
- (2) デバッグ対象 OS でネットワークインタフェースを起動する。
- (3) デバッグ対象 OS 上でパケット受信用のプログラムを動作させる。
- (4) デバッグ支援 OS 上でパケットジェネレータを動作させる。
- (5) 3000 個のパケットを作成し、デバッグ支援機構を呼び出す。
- (6) デバッグ支援機構で、3000 回割り込みを発生させる。
- (7) デバッグ対象 OS の NIC ドライバが割り込みを受けて割り込み処理を開始する。
- (8) 割り込み処理を終えると、カウンタの値をインクリメントする。
- (9) (7) と (8) の処理を 3000 回繰り返した後、デバッグ支援機構がカウンタの値を取得し、割り込みの回数である 3000 回と比較して失敗した回数を求め、この値を返却する。
- (10) パケットジェネレータが割り込み間隔と失敗回数を CSV 形式でファイルに出力する。

## 2.3 パケットのサイズ

サイズとパケットの送信にかかる時間からどの程度の通信量を実現できているかを求める。変化させるパケットのサイズとして、以下のサイズを考えている。

- (1) 64B

ヘッダのサイズが 42B であるため、最小のパケットとして 64B .

(2) 512B

1KB 未満のパケットとして 512B .

(3) 1KB

1KB 以上のパケットの最小サイズとして 1KB .

(4) 1500B

デフォルトで設定されている MTU である 1500B .

(5) 8KB

NIC の受信バッファの最大サイズの半分の 8KB .

(6) 16KB

NIC の受信バッファの最大サイズである 16KB .

### 3 おわりに

本資料では本デバッグ支援環境において、連続でパケットを送信した際に、どの程度の間隔とサイズならば正常にパケット受信処理を行えるかの測定方法について述べた。今後は、この方法で測定を行う。