

# メモリバスの帯域幅の測定

2015/12/17

藤田将輝

## 1 はじめに

デバッグ支援環境を用いた NIC ドライバの性能測定において、パケットサイズが 16KB の場合、30Gbps を実現していると測定した。本デバッグ支援環境では NIC ハードウェアを用いずにパケットの通信を行なっているおり、メモリのみを用いて通信を行なっている。このため、PCI バス等を用いる通信と比べて大きな通信量を実現できていると考えられる。測定した通信量が妥当であることを検証するため、自身の測定環境におけるメモリバスの帯域幅を調査した。具体的には自身の計算機のメモリ規格とこの最大転送量（理論値）を調査した。また、実際にメモリを用いて、データを複写する操作を行うことで、帯域幅を算出した。

## 2 メモリ規格と最大通信量

測定環境におけるメモリ規格は、PC3-10600 DDR3-SDRAM (dual channel) であることが分かった。また、この最大の転送量は 165.6Gbps であった。これらの情報を含めた測定環境を表 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	PC3-10600 DDR3-SDRAM (dual channel) @ 165.6Gbps
NIC ドライバ	RTL8169
ソースコード	r8169.c

## 3 メモリ帯域幅の測定

### 3.1 測定方法

自身の計算機におけるメモリ規格とその最大転送量がわかった。この転送量相当を実現できることを示すため、10  $\mu$  s の間、8KB のデータを複写し続けるシステムコールを作成した。このシステムコールの処理流れ図 1 に示し以下で説明する。

- (1) rdtsc を用いてタイムスタンプを取得する。

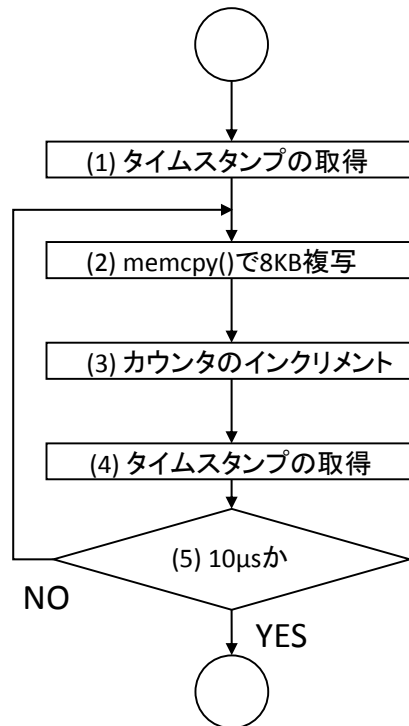


図 1 測定フロー

- (2) ループを開始し，8KB のデータを memcpy() を用いて繰り返し行う．
- (3) memcpy() を終わると，カウンタをインクリメントする．
- (4) rdtsc を用いてタイムスタンプを取得する．
- (5) (1) のタイムスタンプと (3) のタイムスタンプの差分が  $10\ \mu\text{s}$  を超えるとループを抜ける．超えていなければ (2) に戻る．
- (6) カウンタを返却する．

## 3.2 測定結果

返却された値は 22 回であった．したがって，8KB のメモリ複写が 22 回行われたことが分かった．この値と  $10\ \mu\text{s}$  から，測定できたメモリ帯域幅は約 134Gbps であることが分かった．2 章から最大の転送量は 165.6Gbps であることがわかっている．測定したメモリ帯域幅が最大値を超えておらず，際立って小さいこともないため，測定結果は妥当であると言える．

## 4 考察

3 章からメモリの転送量の最大は約 134Gbps であることが分かった．一方，デバッグ支援環境を用いた NIC ドライバの性能測定では，最大で約 30Gbps であることがわかっている．この差は，NIC ドライバのパケット受信処理中で，メモリからパケットを複写する処理以外に長い時間がかかっていることから生じると考えられる．

## 5 おわりに

本資料では、測定環境におけるメモリの最大転送量を調査した。理論値は 165.6Gbps であり、実測値は 134Gbps であった。NIC ドライバにおける最大の転送量は 30Gbps であり、パケット受信処理中でメモリ複写以外に多くの時間がかかっていることが考えられる。今後は、受信処理について詳細に調査し、結果が妥当であることを示す。