

# 共有メモリを用いたデータの受け渡しの確認

2014/9/17

藤田将輝

## 1 はじめに

共有メモリを用いてのデータの受け渡しが成功していることを確認した．本資料では NIC ドライバへの割り込み挿入法の概要，使用する共有メモリの先頭アドレス，および共有メモリを用いたデータ受け渡しの確認について述べる．

## 2 NIC ドライバへの割り込み挿入手法の概要

Mint を用いて，割り込み元 OS から割り込み先 OS の NIC ドライバへ割り込みを挿入する流れを図 1 に示し，以下で説明する．

- (1) 割り込み情報の指定  
プログラマが割り込みジェネレータを用いて割り込み情報を指定する．
- (2) 割り込み情報の通知  
割り込みジェネレータから割り込み元 OS のデバッグ支援機構へ割り込み情報を通知する．
- (3) パケットの生成  
割り込み元 OS のデバッグ支援機構がパケットを生成する．
- (4) 受信バッファへのパケットの格納  
割り込み元 OS のデバッグ支援機構が受信バッファへパケットを格納する．
- (5) 受信バッファ状態の更新  
割り込み元 OS のデバッグ支援機構が受信バッファ状態を受信済み状態へ更新する．
- (6) 割り込み発生要求  
割り込み情報をもとに，割り込み元 OS のデバッグ支援機構からコア 0 へ割り込み発生要求を行う．
- (7) IPI の送信  
割り込み元 OS が保持するコア 0 から割り込み先 OS が保持するコア 1 へ IPI を送信する．
- (8) 割り込み処理の開始  
割り込み先 OS が割り込み処理を開始する．この際，割り込み先 OS が割り込みベクタ番号に対応した割り込みハンドラを実行する．
- (9) 受信バッファの特定  
受信ディスクリプタが保持する受信バッファ状態を割り込み先 OS の NIC ドライバが参照する．これにより，パケットが格納されている受信バッファを特定する．
- (10) ソケットバッファへのパケットの格納

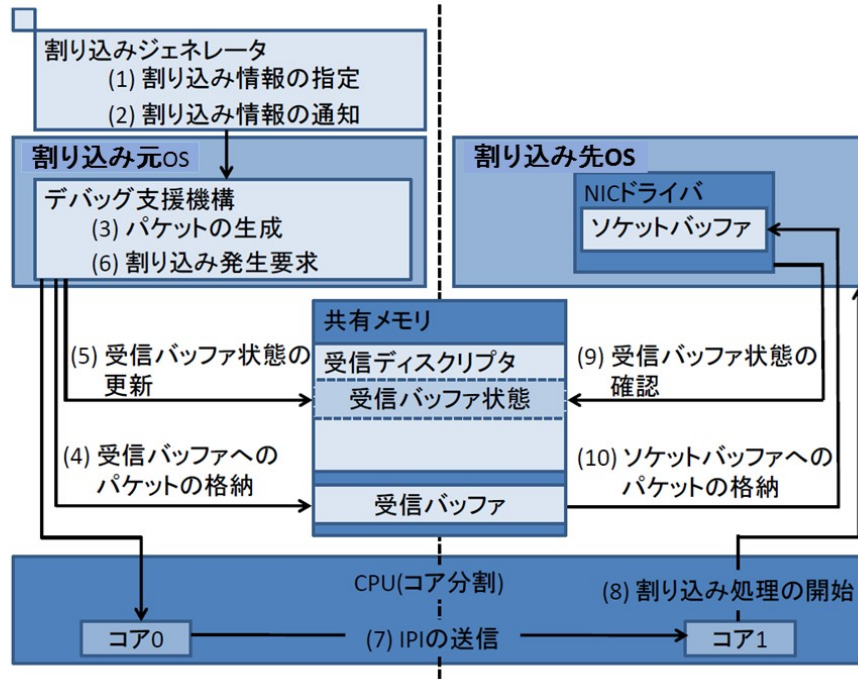


図1 NIC ドライバへの割り込み挿入の流れ

受信バッファからソケットバッファへ割り込み先 OS の NIC ドライバがパケットを格納する。

### 3 使用する共有メモリ領域の先頭アドレス

Mint には共有メモリが用意されている。この様子を図 2 に示し、以下で説明する。具体的には 0x1000000 ~ 0x2000000 が共有メモリとして用意されている。共有メモリ領域の先頭アドレスには Mint のコア管理表のための領域が確保されている。NIC ドライバへのデータを格納する領域を確保するため、コア管理表が使用している領域の終端アドレスを確認する必要がある。このため、コア管理表を作成しているプログラム「mint\_cpumap.c」の中にある、共有メモリを参照している関数の一部をシステムコールとして新たに作成し、調査した。これにより、0x1000018 までを使用していることが分かった。このため、本実験では、0x1000019 以降を使用する。なお、実際にデバッグ支援機構として利用する領域については別途検討する。

### 4 共有メモリを用いたデータ受け渡しの確認

共有メモリを使用し、OS ノード間のデータの受け渡しの確認をするため、簡単な例として、配列に文字列を格納し、データの受け渡しを確認した。具体的には割り込み先 OS で「fujita」という文字列を格納した配列を 0x1000020 に格納し、割り込み元 OS で 0x1000020 から配列に文字列を複写し、これを確認した。どちらもシステムコールにより実現した。0x1000020 は仮の使用アドレスとする。以下に

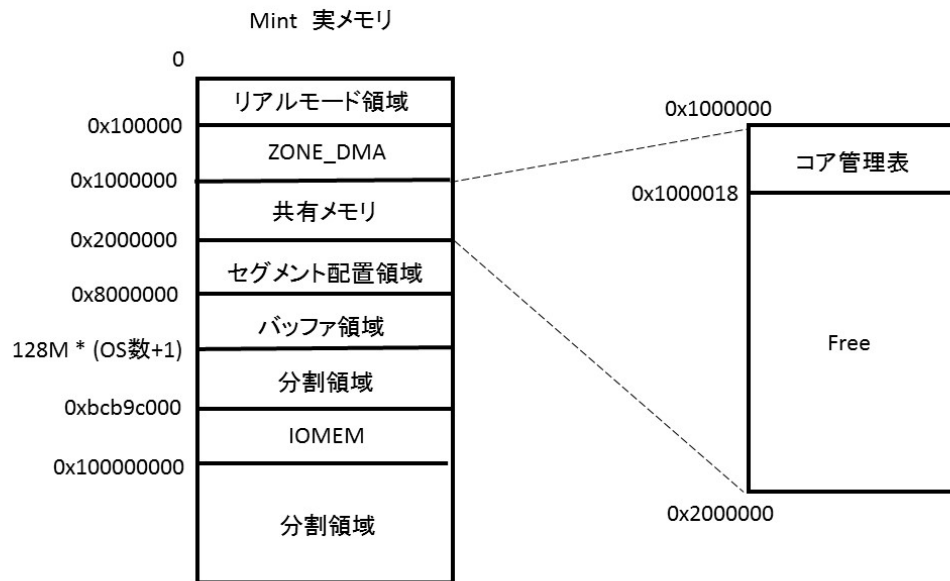


図 2 Mint 共有メモリ

流れを示す。

- (1) 割り込み元 OS でシステムコールを発行する。
- (2) システムコールにより、「fujita」という文字列が格納された配列から文字列を共有メモリの 0x10000020 に格納する。
- (3) 割り込み先 OS でシステムコールを発行する。
- (4) システムコールにより、共有メモリの 0x10000020 から文字列を配列に格納し、標準出力に表示する。

## 5 おわりに

本資料では NIC ドライバへ割り込みを挿入する際のデータの受け渡しについて記述した。割り込み元 OS でデータを共有メモリ領域に格納し、割り込み先 OS で共有メモリに格納されたデータを取得できることを確認した。今後は、使用する共有メモリ領域のサイズ、格納するデータの構造を調査し、本機構における共有メモリ領域の利用法について検討し、実装する。