

Accès direct à la mémoire.

Table des matières

1	Introduction	1
2	User memory transfert	1

1 Introduction

Direct Memory Access. Accès direct à la mémoire. Méthode de transfert de données dans un ordinateur évitant d'avoir à utiliser le processeur et/ou une zone d'E/S standard. Le processeur lance donc le transfert DMA et peut continuer de faire des calculs indépendants (surtout grâce à sa mémoire cache), les périphériques ayant leur propre canal DMA évitent de faire la queue pour l'accès à la zone d'E/S. En fait, ceci est de moins en moins utilisé, car les processeurs sont désormais nettement plus rapides que le reste de la machine, ce qui n'était pas le cas quand le DMA a été mis au point.

Le rapport avec la choucroute :

The PCI-to-ISA bridge, frequently referred to as the South Bridge, connects the root PCI bus to the ISA (or EISA) bus. The South Bridge also typically incorporates the Interrupt Controller, IDE Controller, USB Host Controller and the DMA Controller.

Linux :

```
$ cat /proc/dma
```

2 User memory transfert

```
> For the moment I read cPCI data into a kernel buffer, do byte-swap
> then copy the kernel buffer into the user buffer, which is very time
> consuming.
>
> Is it easy to program the DMA to do the transfer directly from cPCI
> into user memory (the user then has to do byte-swap), that is when
> the blocks are greater than a memory page ?
```

It is not really easy, but the kernel provides an API for that. You have to use `get_user_pages()`, `PageReserved()`, `SetPageDirty()` and `page_cache_release()` to do that. Please have a look at the implementation of `XPC_DMA_EXEC` in `xpc_ioctl_dma()` in `xpc_dma.c` for an example of how to do that.

Byte-swapping in should be fast, too. The PowerPC processor has special load/store instructions for that which are just as fast as the non-swapped variants. All you have to do is use some inline-asm.