

Paginación en multitarea

por Darío Alpern

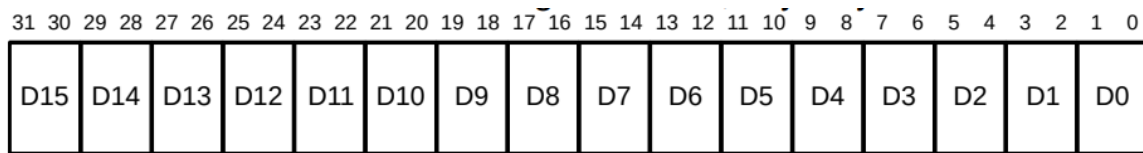
Cuando se debe implementar un sistema multitarea, hay varios ítems para tener en cuenta:

- Existe código y datos con nivel de privilegio de usuario y supervisor (llamados “client” y “manager” respectivamente en la literatura ARM).
- El código y los datos de nivel usuario de una tarea no deben ser accedidos desde otra tarea.

El procesador ejecuta en el nivel de privilegio client cuando los cinco bits menos significativos del registro CPSR indica que está corriendo el modo usuario. Cuando el procesador corre en otro modo (abort, supervisor, system, etc.), el nivel de privilegio es manager.

Para poder indicar el nivel de privilegio que corresponde a una página, se usan los dominios. Existen 16 dominios que se eligen en los bits 8 a 5 de la tabla de primer nivel de paginación.

El coprocesador 15 tiene un registro denominado DACR, cuyo formato es:



DR_n Define los permisos de acceso. El valor n es 0 a 15.

00 No Access. Su acceso generará Domain Fault.

01 Client*.

10 Reservado. Efecto impredecible.

11 Manager*.

El acceso al registro DACR se logra mediante las siguientes instrucciones ejecutando en nivel de privilegio manager:

MRC p15, 0, <Rt>, c3, c0, 0 /* Rt <- DACR */

MCR p15, 0, <Rt>, c3, c0, 0 /* DACR <- Rt */

Con respecto al manejo de traducciones que pertenecen a diferentes procesos, ARM usa el concepto de “Address Space Identifier” (ASID) en el que cada proceso usa un número diferente elegido por el sistema operativo.

De esta manera, una traducción incluye la dirección virtual y el ASID y la unidad de paginación convierte eso en dirección física.

El ASID actual (del proceso que está corriendo) reside en un registro del coprocesador 15 dentro del registro CONTEXTIDR.

Los bits 31 a 8 de CONTEXTIDR contienen PROCID (Process ID), mientras que los bits 7 a 0 contienen el ASID.

Para acceder al registro CONTEXTIDR se utilizan las siguientes instrucciones:

```
MRC p15, 0, <Rt>, c13, c0, 1 /* Rt <- CONTEXTIDR */
```

```
MCR p15, 0, <Rt>, c13, c0, 1 /* CONTEXTIDR <- Rt */
```

Las páginas que usen ASID deben poner el bit nG (non-global) a uno. Ese es el bit 11 de la entrada de la tabla del primer nivel de paginación cuando se usan páginas de 1MB o del segundo nivel de paginación para páginas de otro tamaño.

Las páginas que se usan en todos los procesos (generalmente código y datos de kernel), deben poner el bit nG a cero.

En el cambio de tarea, se debe modificar el valor del registro CONTEXTIDR para que cada tarea use su propio valor de ASID.

Las operaciones necesarias para modificar TTBR0 cuando ocurre un cambio de contexto son las siguientes:

- 1) Cambiar ASID a cero.
- 2) Ejecutar la instrucción ISB
- 3) Cambiar TTBR0 para que apunte a la tabla de primer nivel que corresponde al nuevo proceso.
- 4) Ejecutar la instrucción ISB nuevamente.
- 5) Cambiar ASID al valor que corresponde al nuevo proceso.

Esto supone que no se usa el valor cero para ASID en ninguna tarea.

Translation lookaside buffer

El “translation lookaside buffer” (TLB) es un caché completamente asociativo que tiene las últimas traducciones de direcciones virtuales a direcciones físicas.

Cuando el procesador necesita hacer una traducción, primero verifica si la traducción está presente en la TLB. En caso afirmativo, usa la dirección física y los atributos de página ya almacenados. En caso contrario, el procesador debe acceder a ambas tablas de paginación y luego almacenar la traducción en el TLB.

El formato del TLB es el siguiente:

ASID	Dirección Virtual	Dirección Física	Atributo página	Atributo caché
⌋	⌋	⌋	⌋	⌋

El campo ASID almacena el valor de ASID cuando ocurrió la traducción, siempre y cuando $nG = 1$.

Los campos de dirección virtual y física incluyen solo los bits necesarios para la traducción. En el caso de páginas de 4 KB, estos campos tienen 20 bits.

El campo de atributo de página contiene los atributos que se encontraban en las tablas de paginación en el momento que se hizo la traducción.

Por último, los atributos de caché indican si la entrada de la TLB es válida y también hay bits adicionales para el algoritmo de LRU (las entradas más viejas se descartan para hacer lugar a nuevas traducciones).

Al modificar el registro TTBR0, el procesador mantiene todas las entradas en el TLB.

Se debe tener en cuenta que el TLB soporta 256 valores de ASID diferentes. Si el sistema operativo está corriendo más de 256 procesos al mismo tiempo, deberá borrar manualmente el TLB cuando ocurra un cambio de tarea.

Existen instrucciones para borrar parcial o totalmente el TLB.