

Solución al Ejemplo 1

Enunciado

;Suma de un número variable de datos almacenados en posiciones
;consecutivas de memoria a partir de la variable SUMDOS.

;Cada dato ocupa una palabra y está representado en binario
;puro.

;El número de datos queda definido por la variable N y el
;resultado se almacena en la variable RESULT.

;Si existe desbordamiento se devuelve -1 en el registro r30, en
;caso contrario se devuelve 0 en dicho registro.

```
LEA: MACRO(ra,eti)
or ra,r0,low(eti)
or.u ra,ra,high(eti)
ENDMACRO
```

;Carga el registro ra con la dirección efectiva definida por la
;etiqueta eti.

;Definición de datos utilizados por el programa

```
org 1000
```

```
N: data 5
```

```
SUMDOS: data 0x80000000,24,0x2fffffff,0,3
```

```
RESULT: data 0
```

```
ejl:  or r30,r0,r0      ;Inicializa resultado a 0
      LEA(r2,N)
      ld r2,r2,r0      ;Carga en r2 el número de elementos
      LEA(r20,SUMDOS)  ;Carga en r20 la dirección del primero
      or r3,r0,r0      ;Inicializa la suma acumulada
bucle: cmp r7,r2,r0     ;Si quedan 0 elementos se acaba
      bbl eq,r7,fin
      ld r4,r20,r0     ;Se carga el elemento en r4
      addu.co r3,r3,r4 ;Se suma el elemento a la suma acumulada
      ldcr r7          ;r7 contiene el registro de estado (PSR)
      bbl 28,r7,desb   ;Si el acarreo es 1 salta a desb para
                      ;poner -1 en r30
      addu r20,r20,4    ;Incrementa puntero a la lista
      subu r2,r2,1      ;Queda un elemento menos
      br bucle         ;Salta a bucle
desb: sub r30,r30,1     ;Carga -1 en r30
      br para         ;Salta a finalizar el programa
fin:  LEA(r20,RESULT)  ;Almacena el resultado en RESULT
      st r3,r20,r0
para: stop            ;Para el simulador
```

El objetivo es que la suma del conjunto de elementos almacenados a partir de la etiqueta `SUMDOS` se almacene en `RESULT`. Para ello realizaremos un bucle que se ejecutará tantas veces como indique el dato contenido en `N` (que se cargará en `r4` en la inicialización) e iremos recorriendo la lista con un puntero (`r20`) que se inicializará con la dirección asociada a `SUMDOS`. La suma de elementos se realizará sobre el registro `r3` que en la fase de inicialización se ha cargado con la suma inicial (0).

El inicio de cada iteración (etiqueta `bucle`) compara el número de elementos que quedan por tratar con 0. Si es así se salta a la etiqueta `fin` para almacenar el resultado contenido en `r3` en la dirección de memoria asociada a `RESULT`. Si quedan elementos por tratar se carga el elemento de la lista en `r4` y se acumula a `r3`. Esta suma se realiza con acarreo, de tal forma que en las dos siguientes instrucciones se carga el registro de estado en `r7` y si el flag de acarreo es 1 se salta a la etiqueta `desb` que carga -1 en `r30` tal y como indica el enunciado. Las dos últimas instrucciones del bucle incrementan el puntero a la lista en una palabra (4 bytes puesto que el direccionamiento es a nivel de palabra) y decrementa `r2` en una unidad indicando que queda un elemento menos por tratar. El final del bucle es el salto incondicional a la etiqueta `bucle`.