

ESC1: Pràctica 1

TREBALL PREVI

LECTURA

Nombres naturals

Els nombres naturals es codifiquen en el sistema posicional binari, on el valor numèric N d'una tira d' n bits que codifica un número natural ve determinat per l'expressió:

$$N = \sum_{i=0}^{n-1} b_i \times 2^i$$

El processador SISA-F té suport per executar les operacions aritmètiques de suma, resta, multiplicació i divisió de nombres naturals `ADD`, `SUB`, `MULU` i `DIVU`, així com la multiplicació en doble precisió `MULHU`. El processador també té tres instruccions específiques de comparació entre nombres naturals: `CMPLTU`, `CMPLEU` i `CMPEQ`. Les comparacions $x > y$ i $x \geq y$ es poden programar a partir de les anteriors invertint l'ordre dels operands, però per simplificar la programació hem definit macros per a totes cinc comparacions: `$CMPLTU`, `$CMPLEU`, `$CMPGTU`, `$CMPGEU`, `$CMPEQ`. No hi ha suport explícit per saber si hi ha sobreiximent en el resultat d'una operació aritmètica, ni tampoc produeix cap excepció. Per tant, si es vol detectar, caldrà programar els algorismes corresponents en ensamblador. La divisió per zero produeix una excepció.

Nombres enters

La codificació de nombres enters que suporta el processador SISA-F és en complement a 2, i hi ha suport per realitzar les quatre operacions aritmètiques `ADD`, `SUB`, `MUL`, `DIV` així com la multiplicació en doble precisió `MULH`. També hi ha suport per a les comparacions `CMPLT`, `CMPLE` i `CMPEQ` (el repertori complet està disponible en forma de macros, com en el cas dels naturals). No hi ha suport explícit per a la detecció d'*overflow* en enters, ni tampoc produeix cap excepció. Per tant, si es vol detectar, caldrà programar els algorismes corresponents en ensamblador. La divisió per zero produeix una excepció.

COMPENSIÓ DE CODI

El programa que hi ha al fitxer `s1a.s` realitza la resta en doble precisió (nombres de 4 bytes) de dos nombres enters, `A1` i `A2`, deixant el resultat a la variable `R`. El codi del programa és el següent:

```
.include "macros.s"

.include "crt0.s"

.data

A1: .long 3
```

```

        A2:    .long -2

        R:     .long  0

.text

main: $MOVEI R5, A1

        $MOVEI R6, A2

        LD R1, 0(R5)

        LD R2, 0(R6)

        SUB R0, R1, R2        ;restem els bits -

        $CMPLTU R3, R1, R2    ;calculem borrow

        LD R1, 2(R5)

        LD R2, 2(R6)

        SUB R4, R1, R2        ;restem els bits +

        SUB R4, R4, R3        ;restem borrow

        $MOVEI R5, R

        ST 0(R5), R0          ;guardem part baixa

        ST 2(R5), R4          ;guardem part alta

        HALT

```

Escriviu en un paper, per a cada instrucció i cada macro del programa, quin és el seu resultat (contingut dels registres i de les posicions de memòria que modifica) en executar la instrucció.

DESACTIVAR L'ALARMA

A cada sessió de laboratori teniu 15 minuts per desactivar l'alarma del vago de tren en el que esteu. Per fer-ho connecteu-vos a Atena (des de l'hora d'inici de la classe fins l'hora +15 minuts) i alla trobareu l'enllaç a Genially, a l'apartat **Laboratori-> SCAPE ROOM Salva el Tren**. Heu d'anar al vago que us toqui, en aquest cas l'últim vago del tren. Alla hi trobareu preguntes sobre formats de nombres (naturals i enters) i operacions bàsiques, així com preguntes sobre el codi anterior. Tingueu en compte que algunes preguntes seran sobre el valor dels registres i/o posicions de memòria en diferents instants del programa anterior, així que cal que el porteu treballat.

TREBALL AL LABORATORI

En aquesta sessió programarem petits codis que treballin amb números naturals i enters. Farem la representació d'alguns números en els respectius formats i utilitzarem les operacions aritmètiques i de comparació per traduir sentències d'assignació i condicionals (*if-then-else*).

1. ESBRINA EL TEU CODI

Abans d'entrar en matèria, cal esbrinar quin codi us pertoca resolent el següent enigma:

$$N = \sum_{i=0}^{n-1} b_i$$

on b_i es cada lletra del nom del vostre grup (**en majúscules i sense espais**) codificada en codi ASCII i $n=4$ es a dir només les **4 primeres lletres** del nom del grup.

Un cop tingueu la N cal que la compartiu amb els altres equips i ordeneu els equips per ordre de N , aquest ordre us servirà per repartir-vos l'últim exercici de la pràctica.

2. RESTA DE NOMBRES ENTERS EN DOBLE PRECISIÓ

El programa que hi ha al fitxer **s1a.s** realitza la resta en doble precisió de dos números enters $A1$ i $A2$, deixant el resultat a la variable R (és el programa de l'exercici del treball previ). En aquesta primera activitat heu de comprovar el seu funcionament.

- Assembleu i munteu el programa. Carregueu-lo al depurador. Observeu que al panell dret apareix [Source: crt0.s] i el codi del fitxer **crt0.s** en comptes del nostre fitxer **s1a.s**. Això és degut a la directiva `.include` que hem escrit al principi del programa. No hi ha problema, executeu "pas a pas" (amb F5) les 9 instruccions inicials. Veureu que s'executa un salt a l'etiqueta `main`, i llavors el panell dret ja mostra el codi del vostre fitxer [Source: s1a.s]. Seguiu executant pas a pas (F5), i comproveu a cada pas els valors que haviu previst en el treball previ. Si no coincideixen, intenta trobar una explicació.
- Localitzeu a memòria l'adreça de la variable R . Feu-ho de 2 maneres: a la vista de Dades (comanda "Go", adreça R) o a la vista de Guaites (afegint una guaita per a R). Comproveu que el seu valor final és 5 (0x00000005). És a dir, si ho visualitzeu en mida byte, els 4 bytes a partir de l'adreça de R han de ser: 0x05, 0x00, 0x00, i 0x00.

A continuació comproveu que el mateix programa també resta correctament nombres naturals, en comptes d'enters. Per a això, modifiqueu els enters $A1$ i $A2$ pels següents nombres naturals de 32 bits: 65538 i 65535, respectivament.

- Editeu el fitxer **s1a.s** i modifiqueu el valor inicial de les variables $A1$ i $A2$ amb els valors següents: $A1=65538$ (0x00010002) i $A2=65535$ (0x0000FFFF).
- Assembleu i munteu el programa. Carregueu-lo al depurador, i executeu-lo "pas a pas" (amb la tecla F5), comprovant a cada pas que els valors són els correctes.

3. MULTIPLICACIÓ I DIVISIÓ DE NOMBRES ENTERS

A continuació, editeu el programa del fitxer **s1b.s** i escriviu en la inicialització de les variables $A1$ i $A2$ la parella de valors que vulgueu provar. Llavors assembleu, munteu i executeu el

programa amb el depurador, executant-lo fins al final. I finalment, comproveu el valor de la variable M o D segons si voleu comprovar el resultat d'una multiplicació o el resultat d'una divisió.

Per exemple, proveu els cassos següents:

1. A1 = -2 ; A2 = +3
2. A1 = -32768 ; A2 = -1
3. A1 = +32767 ; A2 = -32767
4. A1 = +1234 ; A2 = -512

4. TRADUCCIÓ DE CODI D'ALT NIVELL

En aquesta activitat heu de traduir a ensamblador del SISA-F els programes següents escrits en C—. Disposem de 4 codis diferents i ens repartirem la feina entre els grups.

Fent servir la N obtinguda al primer apartat, ordeneu els grups per ordre de N (si encara no ho heu fet) i repartiu-vos els codis següents per ordre. (si hi ha dos grups que tenen la mateixa N ordeneu-los alfabeticament).

CODI 0:

```
int a, b, c, d, e, x;

void main ( ) {
    if (a == b) {
        x = c-d*(e+1);
    }
    else {
        x = (c-d*(e+1)+1)/(a-b);
    }
}
```

CODI 1:

```
int a, b, c, d, e, x;

void main ( ) {
    if (a == b) {
```

```

        x = d*e -(a+2);

    }

    else {

        x = (d*e -(a+2)+c) / (b-a);

    }

}

```

CODI 2:

```

int a, b, c, d, e, x;

void main ( ) {

    if (d == b) {

        x = c-d*(a+1);

    }

    else {

        x = (c-d*(a+1)-5) / (d-b)

    };

}

```

Escriviu el vostre programa en un nou fitxer que s'anomeni **s1c.s**. Assembleu-lo i munteu-lo. Heu d'escriure 3 versions del programa:

- En la primera, els valors inicials són a=1; b=0; c=0; d=2 i e=3. Verifiqueu amb el depurador que el valor final de x: és
 - o -7 (pel CODI 0)
 - o -3 (pel CODI 1)
 - o -4 (pel CODI 2)
- Editeu el programa de nou amb els valors inicials a=2; b=0; c=1; d=2 i e=3. En aquest cas, verifiqueu que el valor final de x és:
 - o -3 (pel CODI 0)

- o -1 (pel CODI 1)
- o -5 (pel CODI 2)

I la tercera versio us donara la clau d'accés al següent vago (a la següent practica de laboratori).

* En aquesta versio el valor inicial de les variables d'entrada sera:

* a=3, b=0, c=5 d=6 i e =2.

LLIURAMENT A ATENEA

Connecteu-vos a Atenea i contesteu el qüestionari del vostre grup que hi trobareu en l'apartat **Pràctica 1 → Lliurament LAB 1** sobre els exercicis 1.1, 1.2 i 1.3, i lliureu el programa **s1c.s**

OBTENIR EL PASSWORD

El que heu de fer per obtenir la clau d'entrada al següent vago es combinar el valor de la variable de sortida x de tots els grups participants. El password sera la suma de les x de tots els grups dividit pel numero de grups. El resultat ha d'estar codificat en hexadecimal. Sort!

$$N = \frac{\sum_{i=0}^{n-1} x_i}{e}$$

On e es el numero d'equips i x_i es la x de cada equip.

Teniu una setmana de temps, fins la propera sessio de problemes del vostre grup.

Enjoy! 😊