

Arquitectura e Ingeniería de Computadores

Ejercicios - Tema 3.

Ejercicio. Disponemos de una arquitectura con capacidad de procesamiento VLIW para tres instrucciones o slots y la posibilidad de predicar todas las instrucciones. Para establecer los valores de los predicados se utilizan instrucciones de comparación (cmp) con el formato (p) p1, p2 cmp.cnd x,y donde cnd es la condición que se comprueba entre x e y (lt, ge, eq, ne). Si la condición es verdadera p1=1 y p2=0, y si es falsa, p1=0 y p2=1. Analiza y detalla el desarrollo del código VLIW para el siguiente fragmento de código, teniendo en cuenta que las comparaciones solo se pueden realizar en el slot 1:

```
if (a < b) {  
    c = c - d;  
    if (c >= 0) {  
        a = b + c;  
        b = a - c;  
    }  
    c = c + d;  
} else if (a > b) {  
    a = b - d;  
    b = a + d;  
}
```

El primer paso es obtener la lista de código VLIW correspondiente al fragmento de código anterior:

1. p1, p2 cmp.lt a, b
2. (p1) sub c, d, c
3. (p1) p3, p4 cmp.ge c, 0
4. (p3) add b, c, a
5. (p3) sub a, c, b
6. (p1) add c, d, c
7. p5, p6 cmp.gt a, b
8. (p5) sub b, d, a
9. (p5) add a, d, b

A continuación, debemos analizar las diferentes dependencias entre las instrucciones anteriores y distribuirlas entre los diferentes slots tal y como sea posible:

Slot 1 (cmp)	Slot 2	Slot 3
1. p1, p2 cmp.lt a, b	-	-
3. (p1) p3, p4 cmp.ge c, 0	2. (p1) sub c, d, c	6. (p1) add c, d, c
7. p5, p6 cmp.gt a, b	4. (p3) add b, c, a	5. (p3) sub a, c, b
-	8. (p5) sub b, d, a	9. (p5) add a, d, b