

Resumen Capitulo #2

Hecho por Fernando

FERNANDO AMADOR GARCIA 26 DE ABRIL DE 2021 21:00

Tipos de datos y sentencias de alto nivel

2.1.1. Modos de direccionamiento del ARM

En la arquitectura ARM los accesos a memoria se hacen mediante instrucciones específicas ldr y str.

1. **Direccionamiento inmediato:** Son variables constantes.
2. **Direccionamiento inmediato con desplazamiento o rotación:** Permiten operaciones intermedias sobre los registros.
3. **Direccionamiento a memoria, sin actualizar registro puntero:** Después del acceso a memoria ningún registro implicado en el cálculo de la dirección se modifica.
4. **Direccionamiento a memoria, actualizando registro puntero:** Después del acceso a memoria ningún registro implicado en el cálculo de la dirección se modifica.

Tipos de datos y sentencias de alto nivel

2.1.2. Tipos de datos

1. **byte**
2. **hword**
short
3. **word**
int
4. **quad**

Punteros: Un puntero siempre ocupa 32 bits y contiene una dirección de memoria.

Vectores: Todos los elementos de un vector se almacenan en un único bloque de memoria a partir de una dirección determinada.

Matrices bidimensionales: Una matriz bidimensional de $N \times M$ elementos se almacena en un único bloque de memoria

Tipos de datos y sentencias de alto nivel

2.1.3. Instrucciones de salto

Las instrucciones de salto pueden producir saltos incondicionales (**b y bx**) o **saltos condicionales**.

Cuando saltamos a una etiqueta empleamos **b**.

Y si queremos saltar a un registro lo hacemos con **bx**.

- o **EQ** (equal, igual)
- o **NEQ** (not equal, igual).
- o **MI** (minus, negativo).
- o **PL** (plus, positivo o cero).
- o **CS/HS** (carry set/higher or same, carry activo/mayor o igual).
- o **CC/LO** (carry clear/lower, carry inactivo/menor).
- o **VS** (overflow set, desbordamiento activo)
- o **VC** (overflow clear, desbordamiento inactivo).
- o **GT** (greater than, mayor en complemento a dos).
- o **LT** (lower than, menor en complemento a dos).
- o **GE** (greater or equal, mayor o igual en complemento a dos).
- o **LE** (lower or equal, menor o igual en complemento a dos).
- o **HI** (higher, mayor).
- o **LS** (lower or same, menor o igual).

Tipos de datos y sentencias de alto nivel

2.1.4. Estructuras de control de alto nivel

Se traducen a ensamblador las estructuras de control de alto nivel que definen un bucle (for, while, . . .), así como las condicionales (if-else).

nos explica que podemos ver el resultado de pasar de C a ensamblador y podemos modificar esto.

Tipos de datos y sentencias de alto nivel

2.1.5. Compilación a ensamblador

En este capitulo se explica que podemos pasar de C a Ensamblador y este acompilado y
