



Principio de Diseño de Repertorio de Instrucciones

Universidad de Sonsonate



Clasificación de las Arquitecturas a nivel lenguaje maquina

Universidad de Sonsonate



Clasificación de las Arquitecturas a nivel lenguaje maquina

- Realizaremos una breve introducción a las arquitecturas a nivel lenguaje maquina, la parte de la maquina visible al programador en cual esta compuesta por los **Repertorios de Instrucciones**
- Los repertorios de instrucciones se pueden clasificar en general siguiendo las cinco dimensiones descritas en la imagen, que están ordenadas aproximadamente por el papel que juegan en diferenciar los repertorios de instrucciones



Clasificación de las Arquitecturas a nivel lenguaje maquina

Almacenamiento de operandos en la CPU	Además de en memoria ¿dónde se encuentran los operandos?
Número de operandos explícitos por instrucción	¿Cuántos operandos son designados explícitamente en una instrucción típica?
Posición del operando	¿Puede cualquier operando de una instrucción de la ALU estar localizado en memoria o deben algunos o todos los operandos estar en la memoria interna de la CPU? Si un operando está localizado en memoria, ¿cómo se especifica la posición de memoria?
Operaciones	¿Qué operaciones se proporcionan en el repertorio de instrucciones?
Tipo y tamaño de operandos	¿Cuál es el tipo y tamaño de cada operando y cómo se especifica?



Clasificación de las Arquitecturas a nivel lenguaje maquina

- El tipo de almacenamiento proporcionado para que los operandos estén en la CPU, en contraposición a que estén en memoria, es el factor distintivo más importante entre las arquitecturas a nivel lenguaje máquina. **(Todas las arquitecturas conocidas tienen algún almacenamiento temporal en la CPU).**
- El tipo de almacenamiento de operandos de la CPU a veces dicta el número de operandos explícitamente nombrados en una instrucción. En una clase de máquinas, el número de operandos explícitos puede variar.



Alternativas de Almacenar Operando en el CPU

Almacena- miento temporal proporcionado	Ejemplos	Operandos explícitos por instrucción ALU	Destino para resultados	Procedimiento para acceder a operandos explícitos
Pila	B5500, HP 3000/70	0	Pila	Introducir y sacar de la pila
Acumulador	PDP-8 Motorola 6809	1	Acumulador	Cargar/ almacenar acumulador
Conjunto de registros	IBM 360, DEC VAX	2 ó 3	Registros o memoria	Cargar/ almacenar registros o memoria

Clasificación de las Arquitecturas a nivel lenguaje maquina

- Las elecciones principales son una pila, un acumulador o un conjunto de registros. Los operandos pueden ser nombrados **explícita o implícitamente**:
- **Los operandos en una arquitectura de pila** están implícitamente en el tope de la pila;
- **En una arquitectura de acumulador** un operando está implícitamente en el acumulador.
- **Las arquitecturas de registros de propósito general** tienen solamente operandos explícitos en registros o en posiciones de memoria. Dependiendo de la arquitectura, los operandos explícitos para una operación pueden ser accedidos directamente desde memoria o puede ser necesario cargarlos primero en el almacenamiento temporal, dependiendo de la clase de instrucción y elección de la instrucción específica

Clasificación de las Arquitecturas a nivel lenguaje maquina

- **Ejemplo:** Mostrar cómo sería la secuencia de código **$C = A + B$** en estas tres clases de repertorios de instrucciones.

Pila	Acumulador	Registro
PUSH A	LOAD A	LOAD R1, A
PUSH B	ADD B	ADD R1, B
ADD	STORE C	STORE C, R1
POP C		

Clasificación de las Arquitecturas a nivel lenguaje maquina

- **Ventajas y Desventajas:**

Tipo de máquina	Ventajas	Desventajas
Pila	Modelo sencillo para evaluación de expresiones (polaca inversa). Instrucciones cortas pueden dar una buena densidad de código.	A una pila no se puede acceder aleatoriamente. Esta limitación hace difícil generar código eficiente. También dificulta una implementación eficiente, ya que la pila llega a ser un cuello de botella.
Acumulador	Minimiza los estados internos de la máquina. Instrucciones cortas.	Como el acumulador es solamente almacenamiento temporal, el tráfico de memoria es el más alto en esta aproximación.
Registro	Modelo más general para generación de código.	Todos los operandos deben ser nombrados, conduciendo a instrucciones más largas.



Almacenando Operando en Memoria

Universidad de Sonsonate

Almacenando Operando en Memoria

- Los registros permiten una ordenación más flexible que las pilas o acumuladores, a la hora de evaluar las expresiones. Por ejemplo, en una máquina de registros, la expresión **(A . B) - (C . D) - (E . F)** puede evaluarse haciendo las multiplicaciones en cualquier orden, lo que puede ser más eficiente debido a la posición de los operandos o a causa de las posibilidades de la segmentación.
- Pero en una máquina de pilas la expresión debe evaluarse de izquierda a derecha, a menos que se realicen operaciones especiales o intercambios de las posiciones de la pila.

Almacenando Operando en Memoria

- Hay dos características importantes de los repertorios de instrucciones que dividen las arquitecturas de **registros de propósito general** o **GPR**. Ambas características están relacionadas con la naturaleza de los operandos para una instrucción lógica o aritmética, o instrucción de la ALU:
- **La primera se refiere al número de operandos** (dos o tres) que pueden tener las instrucciones de la ALU. *En el formato de tres operandos*, la instrucción contiene un resultado y dos operandos fuente.
- En el formato de dos operandos, uno de los operandos es fuente y destino para la operación

Almacenando Operando en Memoria

- **La segunda característica de las arquitecturas** GPR está relacionada con el número de operandos que se pueden direccionar en memoria en las instrucciones de la ALU. Este número puede variar de cero a tres. Todas las combinaciones posibles de estos dos atributos se muestran en la figura que se mostrar a continuación, con ejemplos de máquinas. Aunque hay siete posibles combinaciones tres sirven para clasificar aproximadamente todas las máquinas existentes: ***registro-registro (también llamadas de carga/almacenamiento), registro-memoria y memoria-memoria***

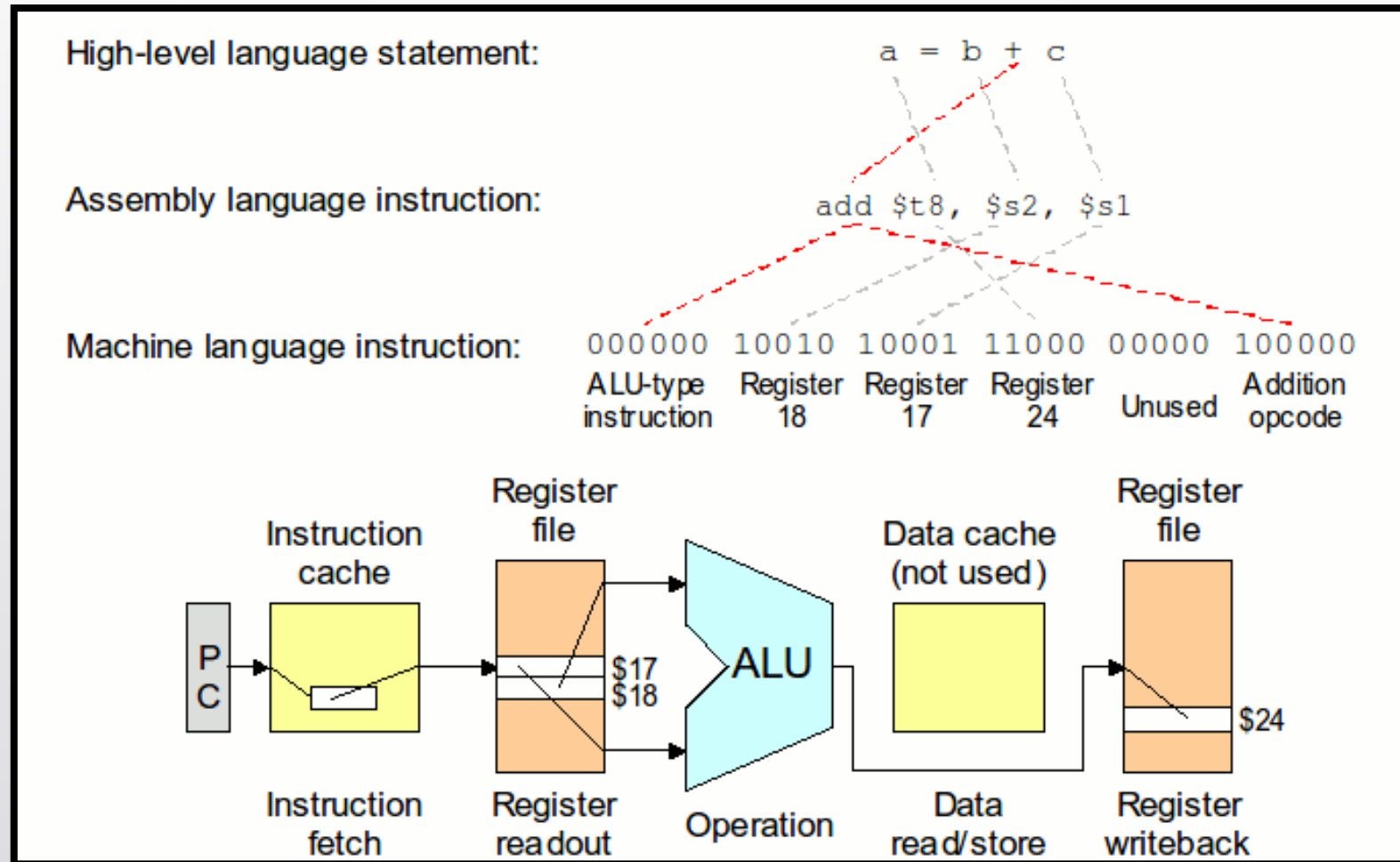
Almacenando Operando en Memoria

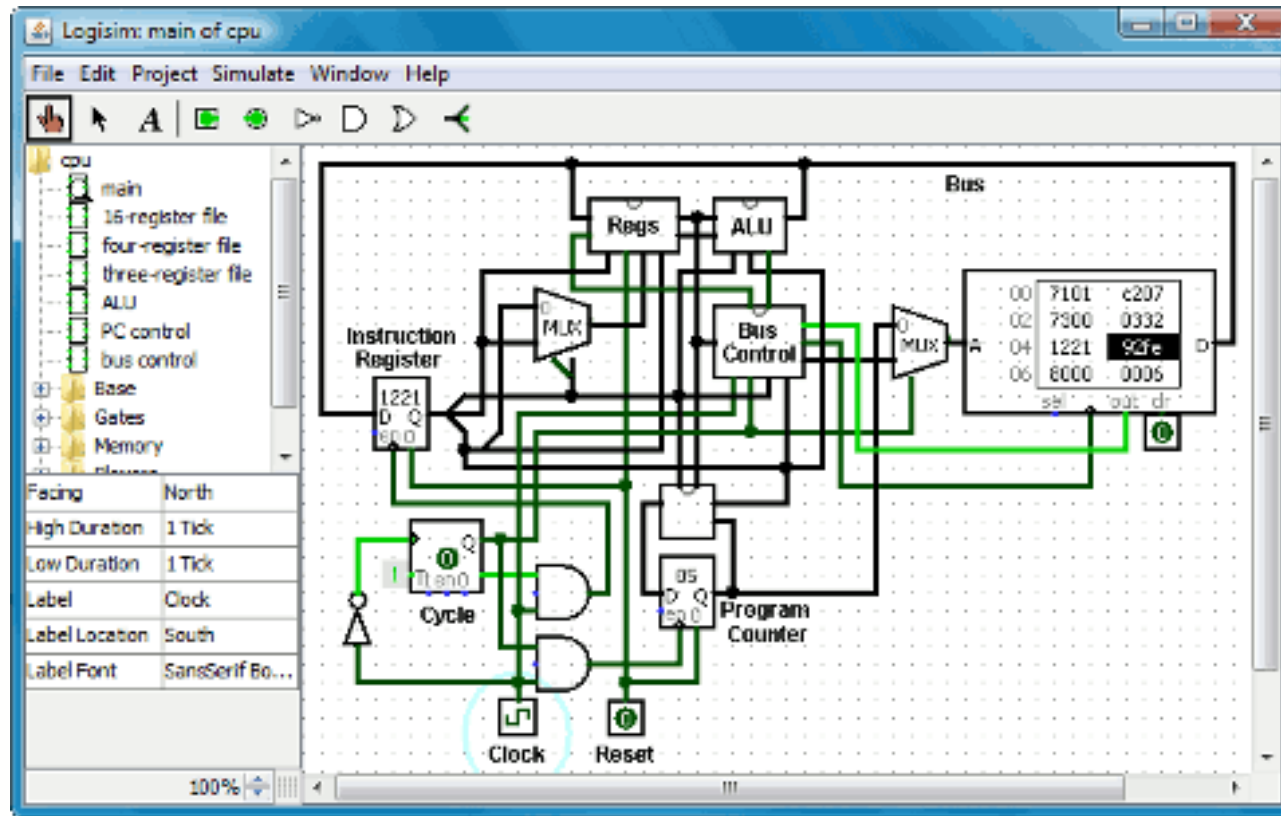
Número de direcciones de memoria por instrucción típica de la ALU	Máximo número de operandos permitidos por instrucción típica de la ALU	Ejemplos
0	2	IBM RT-PC
	3	SPARC, MIPS, HP Precision Architecture
1	2	PDP-10, Motorola 68000, IBM 360
	3	Parte de IBM 360 (instrucciones RS)
2	2	PDP-11, National 32×32, parte de IBM 360 (instrucciones SS)
	3	
3	3	VAX (también tiene formatos de dos operandos)

Almacenando Operando en Memoria

Tipo	Ventajas	Desventajas
Registro-registro (0,3)	Codificación simple de instrucciones de longitud fija. Modelo simple de generación de código. Las instrucciones emplean números de ciclos similares para ejecutarse (ver Capítulo 6).	Recuento más alto de instrucciones que las arquitecturas con referencias a memoria en las instrucciones. Algunas instrucciones son cortas y la codificación de bits puede ser excesiva.
Registro-memoria (1,2)	Los datos pueden ser accedidos sin cargarlos primero. El formato de instrucción tiende a ser fácil para codificar y obtener buena densidad.	Los operandos no son equivalentes, ya que en una operación binaria se destruye un operando fuente. Codificar un número de registro y una dirección de memoria en cada instrucción puede restringir el número de registros. Los ciclos por instrucción varían por la posición de operando.
Memoria-memoria (3,3)	Más compacta. No emplean registros para temporales.	Gran variación en el tamaño de las instrucciones, especialmente en instrucciones de tres operandos. Además gran variación en el trabajo por instrucción. Los accesos a memoria crean cuellos de botella en memoria.

Almacenando Operando en Memoria





Practica con Logisim

Utilización de Clock y Hex
Display