

16 DE FEBRERO DE 2017



PRACTICA 2

INTRODUCCIÓN AL INTERPRETE 80X86 SOBRE LA PLATAFORMA T-JUINO

CARLOS OMAR CALDERON MEZA

FACULTAD DE CIENCIAS QUIMICAS E INGENIERIA - INGENIERO EN COMPUTACION
MICROPROCESADORES Y MICROCONTROLADORES

Objetivo:

El alumno se familiarizará con el intérprete 80X86 (modo 16bits) sobre la plataforma T-Juino.

Esto con el fin de desarrollar programas en lenguaje C para dicha plataforma.

Material:

- Computadora Personal (PC)
- Programa Editor de texto (ASCII), TASM y TLINK
- Tarjeta T-Juino (con intérprete 80x86)
- Manejador FTDI instalado

Equipo:

- Computadora Personal
- Programa emulador de terminal

Teoría:

- Lenguaje C, Lenguaje Ensamblador y combinación de éstos en sistemas embebidos.

Lenguaje c en sistemas embebidos

Los diseñadores tradicionalmente han utilizado lenguajes de programación como C para desarrollar sistemas embebidos basados en arquitectura de microprocesadores o microcontroladores. La popularidad de C ha dado lugar a un gran ecosistema de opciones tecnológicas de herramientas como bibliotecas, compiladores y sistemas operativos.

Si disponemos del lenguaje C, ¿Para qué necesitamos ensamblador? Existen, básicamente, dos razones:

- **Mejora del espacio ocupado y del tiempo de ejecución de un programa:** Los compiladores optimizan muy bien el código, pero en ocasiones, utilizar directamente determinadas instrucciones muy específicas de un procesador mejora de manera considerable estos dos factores. Por ejemplo: stos, lods, etc. en IA32, que permiten copiar una cadena entera con muy pocas instrucciones. Existen diversos ejemplos en `include/asm-i386/string.h`.
- **Programación de sistemas:** Los procesadores actuales disponen de mecanismos que facilitan la gestión de memoria, de procesos y de dispositivos de E/S. Estos mecanismos se utilizan para el desarrollo de los sistemas operativos y el acceso a los mismos desde un lenguaje de alto nivel sería imposible sin herramientas como la directiva **asm**.

La directiva asm

La directiva extendida permite:

- Introducir código en ensamblador.
- Acceder a los diferentes elementos de un programa en lenguaje C (funciones, variables, etc.).
- Informar al compilador de qué elementos del sistema, registros, *flags*, memoria, se modificarán durante la ejecución del fragmento de código en ensamblador y de qué forma. Esto le permite al compilador prever que elementos, generalmente registros, tiene que recargar, guardar en pila, etc.

El formato de la directiva es:

```
__asm__ ( plantillas de instrucciones
          : operandos de salida          /* opcional */
          : operandos de entrada        /* opcional */
          : registros y flags modificados /* opcional */
        );
```

En nuestro caso al utilizar el intérprete T-juino la directiva **asm** es de mucha utilidad ya que no se cargan las librerías de c en el intérprete y se puede acceder a instrucciones del 80x86 directamente implementado funciones con procedimientos de ensamblador.

Desarrollo:

a) Proponer dos números enteros positivos N y M, tal que $M < (N/2)$ y $0 < N < 32$.

b) Seguir los siguientes pasos:

- a. Generar X (el cual es un número de 32 bits) tal que contiene N bits en '1' en su parte menos significativa, y el resto en '0's.
- b. Repetir M veces el siguiente paso, alternando el orden de recorrido, primero empezando del LSB al MSB, después del MSB al LSB y así sucesivamente.
- c. Encontrar el primer bit en '1' y ponerlo en '0', buscar el tercer bit en '1' (ignorando el segundo), repitiendo este paso hasta recorrer todo X.
- d. Imprimir X.

Conclusiones y comentarios:

El lenguaje c por sus características y popularidad encaja perfecto en la programación de sistemas embebidos, casi cualquier microprocesador tiene un compilador de c, programar en el T-Juino se complicó en algunas cosas, primeramente realice la practica en C en un IDE y al pasarlo a la tarjeta varias instrucciones no funcionaron o no las reconoció, por ejemplo el uso de punteros no logre hacer funcionar en la tarjeta, fue necesario implementar el código en funciones con return, a comparación de la primer practica que fue totalmente en asm el código no tuvo problemas a la hora de ejecutar el código en el T-Juino.

Bibliografía:

<http://www.ni.com/white-paper/14623/es/>

<http://server-die.alc.upv.es/asignaturas/PAEEES/2005-06/A07%20-%20Sistemas%20Embebidos.pdf>

<http://web-sisop.disca.upv.es/qii-dso/es/pa5-ensamblador/pa5-ensamblador.html>