

DISEÑO E IMPLEMENTACIÓN DE UN SIMULADOR DE MICROCOMPUTADOR EN UNA PLATAFORMA NO CONVENCIONAL

Ing. Jorge A. León Cano¹

*Facultad de Ingeniería
Universidad Mayor de San Andrés
La Paz—Bolivia*

RESUMEN

Es sabido que en nuestro medio no existe la suficiente cantidad de recursos necesarios para emprender un adecuado estudio de temas relacionados con la tecnología electrónica; es por eso que el trabajo desarrollado brinda una alternativa de solución para reducir las limitaciones en el conocimiento y aplicación de las funcionalidades de los microprocesadores, como partes fundamentales de los microcomputadores.

Se busca definir una línea de conocimiento para el diseño y construcción de simuladores de microprocesadores, multiprocesadores y/o transputers. La concepción del trabajo trae consigo muchos otros tópicos, como por ejemplo la estructuración de ensambladores cruzados y programas monitores para sistemas embebidos.

El caso de estudio del presente trabajo describe el diseño y las características de un sistema de simulación del modelo de programación de microprocesadores MC6800 (como parte del *trainer* ET-3400) haciendo uso de calculadoras programables HP48 del fabricante Hewlett Packard. Se desarrolla una aplicación de software para calculadores que no disponen de un lenguaje de programación estándar bajo los principios de optimización de espacio de almacenamiento, velocidad, usabilidad, modularidad, reutilización, integridad y portabilidad. La concepción del sistema emulador se fundamenta en la aplicación correcta de los conceptos existentes en la arquitectura de hardware y en la ingeniería de software para sistemas empotrados.

1. INTRODUCCIÓN

La implementación de un sistema de simulación en una plataforma como el calculador gráfico de Hewlett Packard es justificable por la funcionalidad de éste, las características de dicho dispositivo permiten, entre otras posibles acciones:

- Recepción y transmisión de datos.
- Visualización de una considerable cantidad de información.
- Procesamiento de la información para el control de eventos.

Es necesario tener un conocimiento apropiado de la arquitectura de hardware y software de éstos calculadores para el desarrollo de aplicaciones relativamente complejas que van más allá de las especificaciones existentes en los manuales de usuario de éstos equipos.

2. ANTECEDENTES

El uso de calculadores programables en diversas aplicaciones ya ha sido abordado dentro de varios proyectos realizados en la Facultad de Ingeniería. Se desarrollaron accesorios para transformar y/o adicionar funcionalidad

al típico calculador programable, pero muchos o casi todos con estructuras poco claras y sin la posibilidad de portabilidad y reuso de los módulos hardware/software.

Con respecto al caso de estudio, ya existen simuladores de microcomputadores implementados en computadores personales o estaciones de trabajo, sin embargo, de acuerdo a la investigación realizada, no se tiene constancia de que se haya desarrollado una herramienta similar para calculadores HP48.

3. OBJETIVOS DE DISEÑO

Desarrollar una aplicación de software para calculadores que no disponen de un lenguaje de programación estándar con el fin de poder simular la operación de microprocesadores en un entorno de entrenamiento.

Para alcanzar el objetivo se tiene que:

- Estudiar y analizar la arquitectura de una plataforma específica como es el calculador HP48 de Hewlett Packard.
- Evaluar los requerimientos para el diseño de un simulador de microprocesadores de entrenamiento del caso de estudio seleccionado, para cuantificar el alcance funcional de la aplicación.
- Diseñar la arquitectura de software, para el caso de estudio seleccionado, considerando los principios de

¹Email: joaleon@hotmail.com

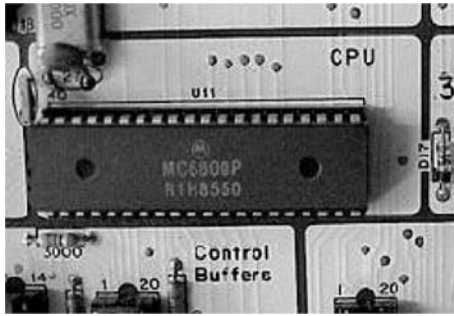


Figura 1. Detalle del MC6800 del *trainer* ET-3400.

optimización de espacio de almacenamiento, velocidad, reusabilidad, modularidad, integridad y portabilidad.

- Implementar la aplicación, para el caso de estudio seleccionado, con el fin de realizar pruebas unitarias, pruebas de módulo y pruebas de integridad de acuerdo a los protocolos estándar de pruebas.
- El proceso de diseño del sistema de simulación no contempla algunas funciones propias del microcomputador real; no se implementan las instrucciones que gestionan las interfaces físicas del *trainer*.
- Extender el desarrollo para realizar la migración a otras plataformas distintas a las del caso de estudio seleccionado.

4. DESCRIPCIÓN DEL DESARROLLO DEL TRABAJO

4.1. *Evaluación de requerimientos*

HARDWARE

Las características de las arquitecturas del *trainer* y el calculador se resumen a continuación:

***Trainer* ET-3400** Microprocesador Motorola MC6800

Ancho de palabra: 8 bits
Espacio direccionable (Máx.): 64 KB
Frecuencia del sistema: 1 MHz aprox.
Registros del modelo de programación: 6

Calculador HP48G+

Microprocesador NEC Saturno
Ancho de palabra: 4 bits
Espacio direccionable (Máx.): 512 KB
Frecuencia del sistema: 4 MHz aprox.
Registros del modelo de programación: 19

Deben resaltarse los siguientes atributos físicos del dispositivo en el que se implemente un simulador de programación de un microcomputador:

MEMORIA. Los calculadores de las familias HP48G+ y HP48GX tienen 128 KB de memoria RAM (sin posibles expansiones), lo que permite el cómodo desarrollo de una aplicación que requiere alrededor de 28 KB.

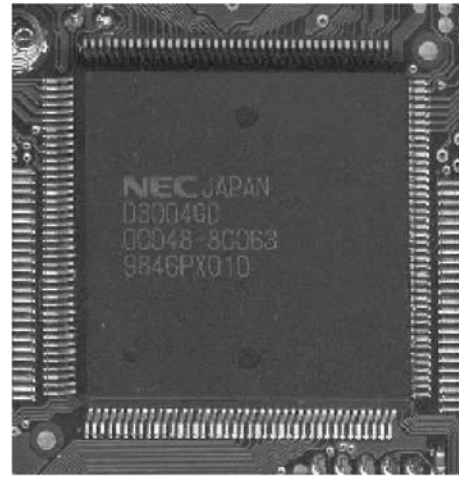


Figura 2. Detalle del μ P Saturno del calculador HP.

VELOCIDAD. Los tiempos de simulación del software implementado deben ser aptos para el estudio de casos de uso y pruebas dentro de un margen de latencia no exagerado. En el calculador existen tres niveles de programación disponibles, en orden descendente respecto al nivel del lenguaje, pero ascendente en cuanto a la velocidad de ejecución se tiene:

- (1) Lenguaje RPL de Usuario (UserRPL).
- (2) Lenguaje RPL del Sistema (SystemRPL).
- (3) Lenguaje Ensamblador (ASM).

INTERFAZ HOMBRE MÁQUINA. En el entorno de simulación, el teclado alfanumérico del calculador es utilizado para imitar al del *trainer*, mientras que la pantalla (resolución 131 x 64) muestra mayor información que la que podemos percibir haciendo uso del *trainer*.

SOFTWARE

El calculador ofrece un adecuado conjunto de instrucciones que permite imitar las operaciones del modelo de programación de la Unidad del Microprocesador MC6800 (UM6800) y el entorno del ET-3400.

4.2. *Diseño de la arquitectura de software*

4.2.1. *Modelado de datos*

Se analizan las características y relaciones funcionales de las variables pertenecientes a los modelos de programación de los sistemas simulado y simulador.

4.2.2. *Reconocimiento del entorno de simulación*

De acuerdo al modelado de datos se asignan correspondencias funcionales entre las variables de simulación del *trainer* y el calculador. Los contenidos de los registros de la UM6800 se almacenan en los registros de la unidad del Saturno. La memoria RAM y ROM de la UM6800 se simulan en la RAM del calculador.

TABLA 1

Correspondencia de registros y memoria de la UM6800 y del μ P Saturno.

UM6800		Saturno	
Registro	Tamaño (bits)	Registro	Tamaño (bits)
Acumulador A	8	R. de trabajo A	64
Acumulador B	8	R. de trabajo B	64
Registro Índice	16	R. de almac. R1	64
Puntero de Pila	16	R. de almac. R2	64
Contador de Programa	16	R. de almac. R3	64
Registro de Código de Condición	8	R. de estado ST	16
Memoria	Tamaño (bytes)	Memoria	Tamaño (bytes)
RAM	512	RAM	512
ROM	1024	ROM	1024

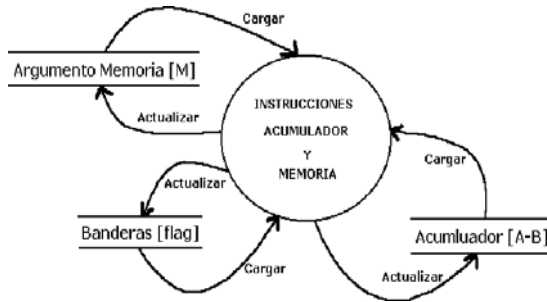


Figura 3. Grafo de las instrucciones de Acumulador y Memoria.

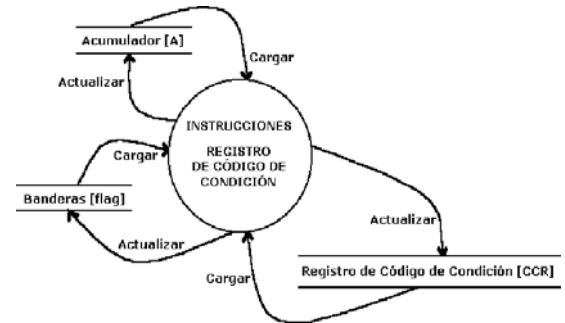


Figura 5. Grafo de las instrucciones del Registro de Códigos de Condición.

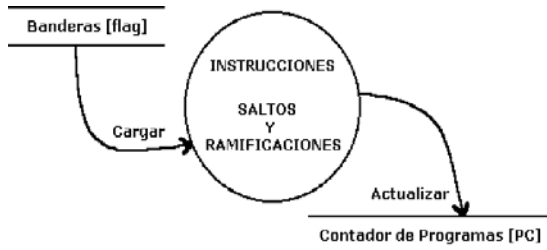


Figura 4. Grafo de las instrucciones de Saltos y Ramificaciones.

4.2.3. Implementación

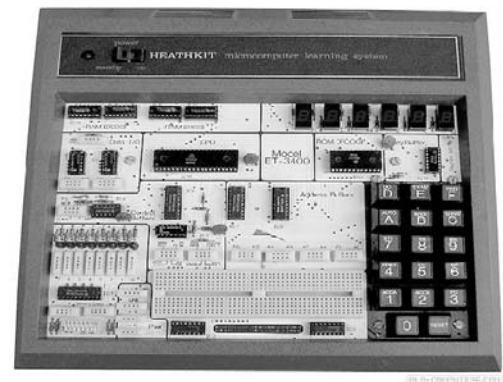
La arquitectura del sistema de simulación se desarrolla refinando sucesivamente niveles de detalle procedimental. Se desarrolla una jerarquía descomponiendo un enunciado macroscópico de función (una abstracción procedimental) al estilo paso a paso hasta que se llega a los enunciados del lenguaje de programación.

Los módulos encargados de la administración de la interfaz gráfica de la aplicación fueron implementados en SystemRPL, mientras que los módulos funcionales responsables de la simulación de la UM680 están escritos en ASM.

Se identifican tres grupos funcionales generales en

el sistema de la UM6800, la representación de éstos se muestra con los grafos del diseño de los modelos estructurales en las figuras 3, 4 y 5.

En la figura 6 se señalan las características físicas del sistema original y las del sistema simulador.

Figura 6. Aspecto del *trainer* ET-3400, fabricado el año 1977.

En la figura 7 se detalla la interfaz hombre máquina del *trainer* real, donde:

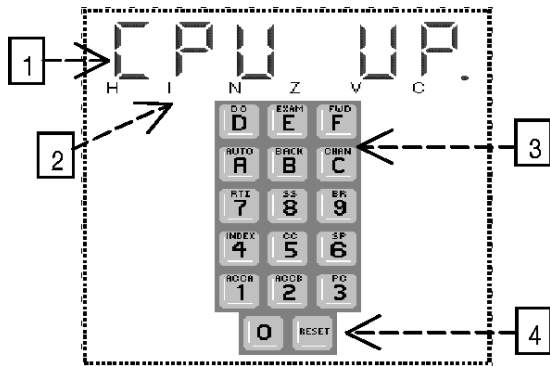


Figura 7. Interfaz hombre máquina del *trainer* ET-3400.

- [1] Indicadores de valor/estado.
- [2] Relación de indicador y bandera.
- [3] Teclado alfanumérico de función/valor.
- [4] Reinicio del programa Monitor.

La pantalla correspondiente al programa de emulación (figura 8) ofrece las mismas características en los indicadores estándar de 7 segmentos. Se muestra la correspondencia entre las banderas y los respectivos indicadores que las representan con el relleno o vacío de las secciones gráficas de cada una de ellas (un cubículo relleno corresponde al valor 1 lógico, y un cubículo vacío corresponde al 0 lógico).

Los valores constantemente actualizados de los acumuladores son los que se muestran en la parte superior derecha. El registro de códigos de condición (CC) muestra la información de los estados de las banderas. También se indican los valores del contador de programa (PC), registro índice (IX) y el puntero de pila (SP).

5. CONCLUSIONES

La concepción del trabajo trae consigo muchos otros tópicos, como por ejemplo la estructuración de ensambladores cruzados y programas monitores para sistemas embebidos.

El adecuado uso de las técnicas de la ingeniería de software para sistemas empotrados permite la opti-

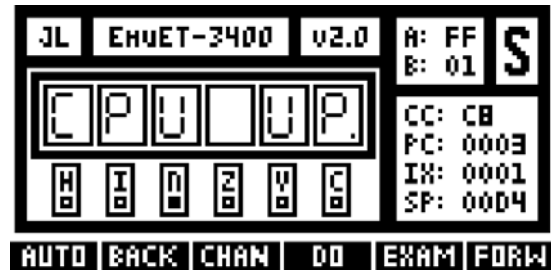


Figura 8. Aspecto de la interfaz gráfica del simulador.

mización del espacio de almacenamiento para datos y la velocidad de simulación.

El uso conjunto de dispositivos con las características del calculador programable de HP y módulos de hardware adicionales pueden ser empleados en varias aplicaciones, como por ejemplo: simulación de eventos y sistemas de hardware/software, control de procesos de mecanismos y telecontrol de eventos (mediante la interfaz infrarroja del calculador), adquisición y análisis oscilográfico de señales, etc.

REFERENCIAS

- [1] Heathkit Educational Systems, "Individual Learning Program in Microprocessors. EE-3401", Heath Company, 1977.
- [2] Hewlett-Packard Company, "The RPL compiler manual", "The Saturn assembler manual", Hewlett-Packard Co., 1991.
- [3] Eduardo Kalinowski, "Programming in System RPL", 1998.
- [4] Matthew Mastracci, "Guide to the Saturn Processor (Whit HP48 Applications)", 1998.
- [5] Roger S. Pressman, "Ingeniería del Software. Un enfoque práctico", McGraw-Hill, 1998.
- [6] Andrew S. Tanenbaum, "Organización de Computadoras. Un enfoque estructurado", Prentice-Hall, 1992.

Información sobre calculadores Hewlett Packard

- [i] <http://www.hpcalc.org>
- [ii] <http://www.hp.com/calculators>

Técnicas para la programación de un emulador

- [iii] <http://www.komkon.org/fms/EMUL8/HOWTO.html>